



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

**MODULY IS VUT PRO VYKAZOVÁNÍ
A KONTROLY PEDAGOGICKÝCH ÚVAZKŮ**

BUT IS MODULES FOR REPORTING AND ACCOUNTING PEDAGOGICAL ACTIVITIES

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. MATĚJ SOJKA

VEDOUcí PRÁCE

SUPERVISOR

Ing. JAROSLAV DYTRYCH, Ph.D.

BRNO 2023

Zadání diplomové práce



146481

Ústav: Ústav počítačové grafiky a multimédií (UPGM)
Student: **Sojka Matěj, Bc.**
Program: Informační technologie a umělá inteligence
Specializace: Informační systémy a databáze
Název: **Moduly IS VUT pro vykazování a kontroly pedagogických úvazků**
Kategorie: Informační systémy
Akademický rok: 2022/23

Zadání:

1. Seznamte se s jazyky PHP a SQL a s technologiemi využívanými při rozvoji webové části centrálního informačního systému VUT.
2. Prostudujte agendu vykazování a výpočtů pedagogických úvazků na FIT VUT a dostupné moduly IS VUT, které lze pro tuto agendu využít.
3. Navrhněte nový modul (či více modulů) webové části IS VUT pro vykazování a kontroly pedagogických úvazků. Zaměřte se při tom na maximální možnou automatizaci procesu vykazování. Součástí řešení bude i osobní přehled úvazků a přehledy úvazků pro vědecké oddělení a vedení fakulty.
4. Implementujte navržené řešení.
5. Zhodnoťte dosažené výsledky a vytvořte stručný plakát prezentující výsledky práce.

Literatura:

- Dle doporučení vedoucího.

Při obhajobě semestrální části projektu je požadováno:

Body 1, 2 a 3.

Podrobné závazné pokyny pro vypracování práce viz <https://www.fit.vut.cz/study/theses/>

Vedoucí práce: **Dytrych Jaroslav, Ing., Ph.D.**

Konzultanti: Ing. Roman Tomšů
Ing. Pavel Witassek
Ing. Jiří Vintera

Vedoucí ústavu: Černocký Jan, prof. Dr. Ing.

Datum zadání: 1.11.2022

Termín pro odevzdání: 17.5.2023

Datum schválení: 31.10.2022

Abstrakt

Cílem této práce je navrhnout a implementovat moduly IS VUT pro vykazování a kontroly pedagogických úvazků pro fakultu informačních technologií, která přechází na celouniverzitní informační systém. Nástroje pro tyto úkony měla vyvinuté v rámci fakulního informačního systému, nicméně měly určité nedostatky. Nové moduly jsou v této práci navrženy se zaměřením na automatizaci činností, které se v rámci nich provádí. Moduly byly navrženy a implementovány do aplikace Teacher, která je součástí IS VUT Web a je vytvořena pomocí frameworku Zend. Práva pro práci s moduly a jejich parametrizace lze upravovat pomocí IS VUT Apollo.

Abstract

The aim of this thesis is to design and to implement modules for the BUT IS for reporting and checking of pedagogical activities for the Faculty of Information Technology, which is moving to a university-wide information system. The tools for these tasks had already been developed within the faculty information system, but had some shortcomings. The new modules are designed in this thesis with a focus on automating the activities performed within them. The modules were designed for the application called Teacher, within the BUT IS Web, and were developed using a Zend Framework. The permissions to work with the modules and their parameterization is configurable through BUT IS Apollo.

Klíčová slova

Výuka, pedagogické úvazky, vykazování úvazků, kontrola úvazků, započitatelné hodiny, informační systém, IS VUT, PHP, Zend, Nette

Keywords

Teaching, pedagogical activities, reporting of activities, checking of activities, creditable hours, information system, BUT IS, PHP, Zend, Nette

Citace

SOJKA, Matěj. *Moduly IS VUT pro vykazování a kontroly pedagogických úvazků*. Brno, 2023. Diplomová práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Jaroslav Dytrych, Ph.D.

Moduly IS VUT pro vykazování a kontroly pedagogických úvazků

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením pana Ing. Jaroslava Dytrycha, Ph.D. Další informace mi poskytli Ing. Roman Tomšů, Ing. Pavel Witassek, Ing. Jiří Vintera a Ing. Marek Strakoš. Uvedl jsem všechny literární prameny, publikace a další zdroje, ze kterých jsem čerpal.

.....

Matěj Sojka
15. května 2023

Poděkování

Rád bych poděkoval mému vedoucímu práce panu Ing. Jaroslavu Dytrychovi, Ph.D., který si pro konzultace se mnou vždy zařídil čas a vždy se mi snažil s řešením mé práce pomoci. Rád bych dále poděkoval kolegům z CVIS, kteří mi vždy s úsměvem vysvětlili všechny záležitosti IS VUT. Zejména bych chtěl poděkovat Ing. Romanu Tomšů, Ing. Pavlu Witasskovi a Ing. Marku Strakošovi.

Rád bych poděkoval mé rodině za to, že mi umožnila možnost tohoto vzdělání a podpořovala mě v něm a při psaní této práce. V neposlední řadě bych chtěl za podporu při psaní této práce poděkovat mé přítelkyni.

Obsah

1	Úvod	3
2	Popis IS VUT	4
2.1	IS VUT Apollo	4
2.2	IS VUT Web	5
3	Analýza pedagogických úvazků na FIT VUT a s nimi spojených částí IS	7
3.1	Popis životního cyklu využívání pedagogických úvazků pro tvorbu platů na FIT VUT	7
3.2	Započitatelné hodiny	8
3.3	Popis práce s pedagogickými úvazky ve WIS	9
3.4	Evidování vykazovaných činností a jejich ohodnocování	9
3.5	Omezení pedagogických úvazků	15
3.6	Analýza akcí prováděných s pedagogickými úvazky	18
3.7	Analýza základních databázových schémat a entit	18
4	Technologie používané pro vývoj IS VUT	20
4.1	Delphi	20
4.2	PHP	20
4.3	Framework Zend	21
4.4	Nette	21
4.5	SQL	21
4.6	Oracle Database	21
5	Návrh	23
5.1	Návrh diagramu případů užití	23
5.2	Návrh databázové struktury pro ukládání pedagogických úvazků	23
5.3	Návrh databázové struktury pro evidenci započitatelných hodin	26
5.4	Návrh získávání dat pro generování pedagogických úvazků	26
5.5	Návrh uživatelského rozhraní modulu pro správu pedagogických úvazků	34
5.6	Návrh uživatelského rozhraní modulu pro zobrazení vykázaných započitatelných hodin	36
6	Implementace	37
6.1	Struktura	37
6.2	Práva	38
6.3	Parametrizace	38
6.4	Modelová vrstva	38

6.5	Správa úvazků	40
6.6	Kontroly	41
6.7	Export započitatelných hodin pro výpočet platů	42
6.8	Sloučení modulů	42
6.9	Generování pedagogických úvazků za semestrální práce a projektové praxe .	43
7	Testování	44
7.1	Testování správnosti generování pedagogických úvazků	44
7.2	Testování pomocí testování identit	45
7.3	Uživatelské testování	45
8	Závěr	48
	Literatura	49
A	Obsah přiloženého paměťového média	51
B	Plakát	52

Kapitola 1

Úvod

Začátkem akademického roku 2022/2023 se studijní agendy Fakulty informačních technologií VUT přesunuly z fakultou vyvíjeného webového informačního systému WIS do celouniverzitního IS VUT, který spravuje Centrum výpočetních a informačních služeb VUT v Brně. Agenda vykazování a kontroly pedagogických úvazků byla jedna z těch, kterou bylo zapotřebí přemístit do nového informačního systému. Bohužel tato agenda nebyla v rámci IS VUT implementována, a proto bylo rozhodnuto o tom, že pro tuto agendu budou vytvořeny nové moduly.

Fakulta informačních technologií potřebovala nástroje pro vykazování jednotlivých pedagogických úvazků v rámci předmětů, nástroje pro kontrolu a nástroje pro zobrazení přehledů pro vědecké oddělení a vedení fakulty. Tato agenda je pro fakultu důležitá, protože právě pomocí této agendy se odměňují vyučující za pedagogickou činnost.

Cílem této práce je prostudovat agendu vykazování a kontroly pedagogických úvazků a vytvořit návrh modulů do IS VUT, které jsou zapotřebí pro to, aby se agenda mohla přemístit do nového informačního systému. Jelikož činnosti, za které se vykazují pedagogické úvazky, jsou z velké části již evidované v IS VUT, tak se tato práce zaměřuje na maximální možnou automatizaci procesů vykazování a kontroly úvazků. Pro dosažení tohoto cíle bylo zapotřebí seznámit se s procesy agendy a s postupy, pomocí kterých byla do nynějška spravována. V druhé řadě bylo nutné seznámit se s IS VUT, technologiemi, pomocí kterých je vytvořen, nástroji, které nabízí, a s daty, která spravuje. V dalším kroku bylo možné navrhnout jednotlivé části modulu a následně na základě návrhu moduly vytvořit, otestovat a opravit nedostatky, které testování odhalilo.

IS VUT spravuje Centrum výpočetních a informačních služeb, které je samostatnou součástí VUT. Z tohoto důvodu byl vývoj modulu konzultován se zaměstnanci centra. První verze modulu byla nasazena již v září akademického roku 2022/2023 a doposud bylo pomocí tohoto modulu vykázáno více než 4000 pedagogických úvazků.

Kapitola 2 se věnuje popisu IS VUT. V kapitole 3 se analyzuje problematika pedagogických úvazků na Fakultě informačních technologií VUT. Kapitola 4 se věnuje popisu technologií, které byly použity pro vývoj IS VUT, a v kapitole 5 se využívá poznatků z předšlých kapitol pro návrh nových modulů. V kapitole 6 je popsána implementace řešení a v kapitole 7 jeho testování. V závěrečné kapitole 8 jsou zhodnoceny výsledky práce.

Kapitola 2

Popis IS VUT

Pro následující kapitoly je důležité, aby čtenář věděl, co je IS VUT, k čemu slouží, z čeho se skládá a jak je zamýšlena práce s jeho jednotlivými částmi. Chtěl bych poděkovat Ing. Markovi Strakošovi, který mi k historii a aktuálnímu stavu IS VUT poskytl nemálo informací a budu z nich vycházet v této kapitole.

Na přelomu tisíciletí byla na VUT situace taková, že pokud si fakulta spravovala nějaké agendy digitálně, tak to ve většině případů dělala pomocí systémů, které si vyvinula pro své potřeby. V této době si vedení univerzity uvědomilo, že je zbytečné, aby jednotlivé fakulty plýtvaly zdroji na vývoji IS pro danou fakultu, a proto se rozhodlo vytvářet celouniverzitní IS. Začal tedy vznikat IS VUT Apollo, se kterým pracovali především zaměstnanci jednotlivých fakult. Časem požadavky na IS VUT rostly a jedním z nich byla nutnost obsluhovat uživatele na zařízeních s různými operačními systémy. I z tohoto důvodu bylo okolo roku 2003 rozhodnuto o vývoji webového přístupu k IS VUT, dnes známého pod názvem IS VUT Web. Oba tyto systémy se souběžně vyvíjely. Z počátku byl upřednostňován vývoj IS VUT Apollo, ale časem se projevila nutnost obsluhovat v IS VUT i jednotlivé studenty, a z tohoto důvodu se začal preferovat vývoj IS VUT Web. Chtěl byl ovšem zmínit, že oba tyto systémy jsou vytvořeny za jiným účelem a dovolují svým uživatelům s IS VUT pracovat tak, aby jim jejich práci co nejvíce zefektivnily.

Kapitola je uspořádána následovně. V první podkapitole (2.1) je popsán IS VUT Apollo a v druhé podkapitole (2.2) je popsán IS VUT Web a jeho části.

2.1 IS VUT Apollo

IS VUT Apollo je desktopovou aplikací vytvořenou v grafickém vývojovém prostředí Delphi (4.1), která slouží pro práci s IS VUT. Tato aplikace je primárně vyvinuta pro zaměstnance fakult, kteří v ní pracují s daty týkajícími se studia, vědy a výzkumu nebo administrace IS VUT. Aplikace uživatelům obecně umožňuje velmi svobodnou manipulaci s daty, jelikož spoléhá na to, že uživatelé s přístupem do této aplikace jsou na tuto činnost dostatečně kvalifikováni a ví, jak data v systému mohou a nesmí upravovat, například vzhledem ke směrnici dané fakulty.

Pro administraci IS VUT pak aplikace poskytuje následující dva hlavní nástroje. Prvním je nástroj pro správu práv. Právo v IS VUT může být přiděleno uživateli a umožnit mu získat přístup ke konkrétním datům nebo mu umožnit provádět konkrétní operaci nebo skupinu operací, které jsou na dané právo navázány. Právo může být omezeno rolí. Role je specifikována kolekcí identifikátorů organizačních jednotek, na jejichž data může být apliko-

váno toto právo. V tomto kontextu tedy role pouze omezuje, na která data může být právo použito. Pro seskupení práv a zjednodušení jejich správy se používají profese, ke kterým lze přiřadit množinu práv, která budou uživatelé s danou profesí sdílet. Přiřazování uživatelů k profesím lze automatizovat pomocí databázových dotazů, které lze s profesí asociovat.

Druhým nástrojem je parametrizace, která slouží pro specifikaci nastavení modulů v IS VUT Web a IS VUT Apollo. Administrátorovi tento nástroj umožňuje u modulu pro danou fakultu nastavit jeho viditelnost a hodnoty parametrů, kterými ovlivní, jak se modul bude zobrazovat nebo chovat. Tento nástroj se stává užitečným v momentě, kdy jednotlivé fakulty vyžadují rozdílné chování pro stejný modul nebo jeho periodické úpravy v průběhu školního roku.

V IS VUT Apollo se nachází také nástroj pro dokumentaci databázových schémat. Jedná se o systém, ve kterém lze vytvořit a spravovat entitně vztahové modely IS VUT, vyhledávat v nich řetězce a přidávat k nim poznámky. Užitečnou funkcí, která výrazně zjednodušuje analýzu struktury databáze, je schopnost tohoto nástroje z databáze získat hodnoty číselníků a ty uživateli zobrazit.

2.2 IS VUT Web

IS VUT Web je webová aplikace vytvořená pomocí programovacího jazyka PHP, která oproti IS VUT Apollo nabízí způsob interakce se systémem i studentům a uchazečům o studium. Celá webová aplikace se skládá z několika menších aplikací, které sdílí základní implementaci. Jako příklad sdílené implementace by se dalo uvést přihlašování do systému, které se napříč aplikacemi v rámci IS VUT Web neliší. Celý systém taktéž využívá stejnou databázi jako IS VUT Apollo, a to Oracle Database, která je distribuovaná pomocí nástroje Oracle RAC.

Aktuálně IS VUT Web nabízí webové rozhraní pro veřejnost a spolu s ním jeho redakční systém pro administrátory a aplikace StudIS, Teacher, Intraportál, E-přihláška a E-learning. Řada poskytovaných nástrojů v rámci systému se neustále udržuje a rozšiřuje. Jako příklad lze uvést nedávné předělávání API a záměr v blízké budoucnosti vytvořit aplikaci, která bude napříč univerzitou slučovat agendy věnující se vědě a výzkumu.

StudIS je nejvytíženější aplikací IS VUT Web a slouží studentům, kterým nabízí informace, které potřebují v rámci svého studia. Jedná se taktéž o první aplikaci, která byla v rámci tohoto systému vyvíjena. Její první vývoj probíhal v programovacím jazyce PHP ve verzi 3. Aktuální verze PHP, na které běží, je 7.4.

Druhou velkou aplikací v rámci IS VUT Web je Teacher. Aplikace Teacher slouží zejména vyučujícím, případně studijním poradcům nebo referentkám na fakultě. Aplikace uživatelům umožňuje činnosti jako je vypisování zkoušek, zadávání hodnocení nebo například editace rozvrhů. První verze byla napsána v PHP 3 a postupem času byla aplikace reimplementována ve frameworku Zend ve verzi 1 v kombinaci s šablonovacím systémem Dwoo. Před přechodem IS VUT Web na PHP ve verzi 8 je v plánu aplikaci Teacher přeprogramovat do frameworku Nette, aby se sjednotily použité nástroje s novými aplikacemi a zjednodušila se tak údržba systému.

Aplikaci Intraportál naopak mohou využívat všichni uživatelé IS VUT Web. Součástí této aplikace je totiž požadavkový systém, pomocí kterého mohou zaměstnanci spravovat interní požadavky. Pomocí aplikace Intraportál lze taktéž například vytvořit rezervaci místnosti mimo výuku.

Aplikaci E-learning mohou využít učitelé pro výuku předmětu. Mohou zde zveřejňovat materiály k předmětu, tvořit online testy anebo aplikaci použít pro hromadnou komunikaci

se studenty. V rámci aplikací IS VUT Web se často používá pojem modul. Modulem se v tomto kontextu chápe konkrétní záložka z navigačního menu v konkrétní aplikaci.

Jak již bylo zmíněno, aplikace Teacher používá framework Zend a ten využívá architekturu MVC. Typický modul v aplikaci Teacher má svůj dedikovaný kontroler, ve kterém jsou definovány všechny akce, které podporuje. Pro popis vzhledu stránky se využívá šablonovací nástroj Dwoo a všechny soubory ve formátu `.phtml` popisující jednotlivé pohledy se ukládají ve složce určené pro tyto soubory daného kontroleru. Komponenty stránky, jejichž využití se v rámci kontroleru opakuje, se zapisují do zvláštního souboru, ze kterého se poté do jednotlivých pohledů importují. Definice formulářů pro modul se zapisuje taktéž do specifické složky určené pro formuláře kontroleru. V rámci formulářů je běžnou praxí, že se základní implementace některých formulářů sdílí pro lepší udržitelnost mezi několika moduly. K datům v databázi se přistupuje s pomocí modelů, které jsou dostupné v knihovně, která je dostupná všem aplikacím IS VUT Web. Každý model definuje atributy modelu, metody pro získání dat z databáze, metody pro namapování dat na atributy, metody pro práci se seznamem prvků tohoto typu a metody specifické pro daný model. V nové verzi modelu, která se aktuálně začíná používat, přibyla nad těmito metodami ještě byznys vrstva, která má za úkol byznys logiku spojenou s modelem udržovat izolovanou od implementace kontroleru, aby ji bylo možné znovu využít při implementaci jiné aplikace.

Kapitola 3

Analýza pedagogických úvazků na FIT VUT a s nimi spojených částí IS

Fakulta informačních technologií VUT v Brně (dále jen FIT) v září roku 2022 přešla z fakultou vyvinutého informačního systému WIS FIT na celouniverzitní systém IS VUT. V rámci přechodu fakulty na nový informační systém bylo třeba najít způsob, jakým agendy, pro které fakulta využívá WIS, začít obsluhovat v IS VUT. Většina, pro vysokou školu běžných agend, měla v IS VUT již existující implementaci, a proto je bylo možné bez větších problémů přemístit do IS VUT. Příkladem agendy, která neměla v IS VUT odpovídající nástroje v době rozhodnutí a přechodu fakulty na nový informační systém, je agenda pro vykazování a kontrolu pedagogických úvazků.

Vykazování a kontrola pedagogických úvazků slouží fakultě k evidenci pedagogických činností, za které pak může vyplácet odměny. Vykazování úvazků může fakultě taktéž sloužit jako nástroj pro monitorování toho, jakým způsobem je na fakultě praktikována výuka nebo k rychlému zjištění toho, jaký by mělo finanční dopad finanční zvýhodnění nepopulárního, ale vedením fakulty preferovaného způsobu výuky. Rozhodnutí o tom, jakým způsobem se pedagogické úvazky mají v rámci fakulty počítat, specifikovalo vedení fakulty ve směrnici č. 4/2019 [10], která se pro mě stala hlavním zdrojem informací při psaní této kapitoly. Pokud není v textu uvedeno jinak, tak informace v této kapitole čerpám z tohoto dokumentu.

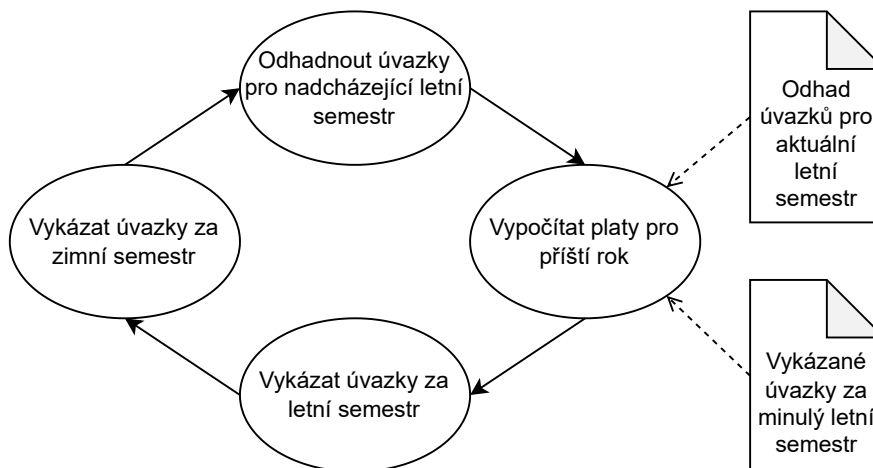
V této kapitole je popsáno, jak se pracuje s pedagogickými úvazky a jak se doposud vykazovaly a ohodnocovaly. Dále jsou v kapitole popsány části IS VUT, které souvisí s pedagogickými úvazky a činnostmi s nimi spojenými.

3.1 Popis životního cyklu využívání pedagogických úvazků pro tvorbu platů na FIT VUT

Na konci zkuškového období zimního semestru je zapotřebí, aby vyučující měli vykázanou pedagogickou činnost za zimní semestr a vytvořili kvalifikovaný odhad výkazu pedagogické činnosti na nadcházející letní semestr. V průběhu letního semestru pak děkan fakulty tyto informace o pedagogických úvazcích v letním a zimním semestru využije pro výpočet platů, kterými budou pedagogičtí pracovníci odměňováni v průběhu dalšího školního roku. Po ukončení letního semestru je opět zapotřebí, aby vyučující vykážali pedagogickou činnost,

kteřou prováděli v průběhu letního semestru, a napravili tak chyby, kterých se mohli dopustit ve svých odhadech pedagogické činnosti na letní semestr. Celý tento proces je pro názornost ještě znázorněn na diagramu 3.1.

Důležité je dodat, že v rámci výpočtu platu na další rok se bere v potaz chyba, které se dopustil vyučující v minulém roce ve svém odhadu pedagogické činnosti na letní semestr. Pro vyučujícího je nejvýhodnější se ve svém odhadu neplést. Pokud se ovšem ve svém odhadu spletl, je pro něj výhodnější, když odhadl menší množství práce za letní semestr, než byla realita.



Obrázek 3.1: Schéma popisující životní cyklus využívání pedagogických úvazků pro výpočet platů.

3.2 Započítatelné hodiny

Započítatelné hodiny jsou pro fakultu jakousi jednotkou reprezentující finanční hodnotu odvedené pedagogické činnosti. Za jejich řádnou evidenci odpovídají garanti předmětů, ve kterých je činnost vykonána [10].

Jak již bylo zmíněno, započítatelné hodiny slouží fakultě pro výpočet platů vyučujících. Dále jsou ještě použity pro specifikaci množství pedagogické činnosti, kterou musí student doktorského studia vykonat, aby byl připuštěn ke státní doktorské zkoušce. Činnost v rámci limitu minimálního množství započítatelných hodin, který je jako student povinen získat, mu fakulta nemusí proplácet [9].

$$ZH = (k_1 + k_2 * S) * H \quad (3.1)$$

Koncept započítatelných hodin byl navržen s myšlenkou toho, aby se dal relativně jednoduše zobecnit a implementovat do informačního systému. Obecný zápis výpočtu je znázorněn v rovnici 3.1, kde ZH je počet započítatelných hodin, S je počet studentů, H je počet hodin, které si vyžaduje uskutečnění popisované činnosti. Počet hodin H může být v rámci některých činností považován za nepodstatný a z rovnice vynechán. k_1 je konstantou reprezentující fixní část ZH za činnost, zatímco k_2 je konstantou ovlivňující vliv počtu studentů na ohodnocení činnosti.

Pro volitelné a povinně volitelné předměty směrnice dále specifikuje, že započítatelné hodiny mohou být vykázány pouze v případě, když předmět, v rámci kterého byla prová-

Jazyk výuky	Typ studia	Minimum zapsaných studentů
Čeština	Bc. studium	20
	Mgr. studium	10
Angličtina	Bc. a Mgr. studium	5

Tabulka 3.1: Závislost jazyka výuky volitelného nebo povinně volitelného předmětu, typu studia, v rámci kterého je vyučován, a minimálního počtu zapsaných studentů potřebných pro to, aby bylo možné v akademickém roce za předmět vykázat úvazky.

děna pedagogická činnost, splňuje podmínky minimálního počtu zapsaných studentů (viz tabulka 3.1), a to 3 dny před začátkem výuky v semestru anebo někdy později v průběhu semestru. Pokud je předmět otevřen a před začátkem semestru nebo v jeho průběhu nenaplní požadavky na minimální počet zapsaných studentů, tak garant předmětu nemůže za daný semestr vykazovat úvazky. Podlimitní počet zapsaných studentů lze opakovaně přenést v rámci předmětu do dalšího roku. V momentě, kdy počet zapsaných studentů a počet přenesených zapsaných studentů naplní podmínku pro minimální počet studentů, učitel může připsat počet přenesených studentů k činnostem prováděným v aktuálním školním roce.

3.3 Popis práce s pedagogickými úvazky ve WIS

V informačním systému WIS, který fakulta donedávna využívala, byl proces zadávání úvazků implementován následujícím způsobem. Člověk pověřený správou předmětu (ve většině případů garant) měl k dispozici tabulku vázající se na vyučovaný předmět v semestru. Tuto tabulku mohl postupně zaplňovat údaji o tom, jaká činnost byla provedena, kdo ji provedl a případně kolik hodin činnost trvala, kolik studentů zahrnovala a kolikrát se opakovala. Ke každému záznamu bylo možné přidat komentář, ve kterém mohl autor vysvětlit, proč vyplnil výkaz zrovna tímto způsobem a zjednodušit tak proces kontroly výkazu.

V momentě, kdy měly být úvazky zkontrolovány, mohla fakultou pověřená osoba spustit automatické kontroly, které u jednotlivých předmětů poukázaly na některé nesrovnalosti ve výkazu. Po automatické kontrole mohla začít kontrola manuální, kdy pověřená osoba kontrolovala výkazy jednotlivých předmětů a zjišťovala, zda se v nich nevyskytují chyby a případně zda automaticky nahlášené chyby nejsou nějakým způsobem odůvodnitelné. Pokud byl záznam o činnosti v pořádku, kontrolující osoba nastavila záznam jako zkontrolovaný. V případě nálezu nesrovnalostí bylo zapotřebí informovat garanta předmětu a počkat, než chyby napraví, a následně záznam zaevidovat jako zkontrolovaný. V momentě, kdy byly všechny záznamy nastaveny jako zkontrolované, byl proces kontroly ukončen.

V průběhu celého procesu vykazování a kontroly pedagogických úvazků bylo možné ze systému vyexportovat informace o vykázaných úvazcích, aby je bylo možné použít pro výpočet platů vyučujících, jak bylo zmíněno v podkapitole 3.1.

3.4 Evidování vykazovaných činností a jejich ohodnocování

Započítatelné hodiny fakultě slouží jako jednotka odpracované práce, která může být proplacena. Na tuto jednotku je nutné převést různé činnosti. Za účelem automatizace procesu vykazování pedagogických úvazků je třeba mít informaci o tom, jestli a jak vyučující evidují činnosti, které jim fakulta může proplatit, v IS VUT. Ne všechny činnosti jsou si z hlediska

Specifikace aktivity			ZH
Činnost	Typ studia	Popis	
Přímá pedagogická činnost	Bc. a Mgr. studium	Přednášky	$(4 + 0,01 * S) * H$
		Demonstrační cvičení	$(3 + 0,01 * S) * H$
		Cvičení a laboratoře	$(2 + 0,01 * S) * H$
		Obhajoby projektů	$0,10 * S * H$
		Individuálně hodnocené projekty	$0,09 * S * H$
		Automatizovaně hodnocené projekty	$0,08 * S * H$
		Kruhové konzultace	2
	Prezenční doktorské studium	Přednášky	$(4 + 0,01 * S) * H$
		Cvičení a laboratoře	$(2 + 0,01 * S) * H$
		Vedení řízeného studia	$15 * S$

Tabulka 3.2: Tabulka popisující činnosti spojené s přímou pedagogickou činností a výpočty započitatelných hodin (ZH) s nimi spojenými, kde S je počet studentů, H je počet hodin.

náročnosti rovny, a proto se používá obecná rovnice (3.1), pomocí které se specifikují výpočty započitatelných hodin za jednotlivé činnosti.

Výpočty úvazků pro jednotlivé činnosti jsou ve směrnici [10] rozděleny do dvou hlavních skupin. První skupina přímé pedagogické činnosti a druhá skupina obsahuje činnosti spojené s vedením, oponováním a obhajobami. Pro přehlednost jsem se v rámci své práce rozhodl činnosti rozdělit na více menších skupin, podle toho, jak spolu souvisí v kontextu práce v informačním systému. Přehledy výpočtů úvazků pro jednotlivé činnosti ve skupinách byly sepsány do tabulek 3.2, 3.3, 3.4, 3.5, 3.6 a 3.7.

První činnosti, kterým se budu věnovat, jsou spojené s přímou pedagogickou činností (tabulka 3.2), konkrétně s přednáškami a cvičeními. Po vytvoření předmětu v IS VUT může garant nebo jiná pověřená osoba vložit do systému položku rozvrhu, u které může specifikovat její typ, kapacitu studentů, učebnu a časové okno, které zabírá. Takovéto položce se v IS VUT říká vyučování. Typ vyučování pak specifikuje, zda se jedná například o přednášku, demonstrační cvičení, cvičení nebo laboratorní cvičení. U vyučování v rozvrhu je taktéž zapotřebí nastavit, v kterých týdnech výuky se bude konat. U přednášek je běžné, že se konají každý týden, ale u cvičení se může stát, že se bude konat například pouze třikrát za semestr. Na přednášky a demonstrační cvičení na FIT VUT se studenti bakalářského a magisterského studia nemusí registrovat, a tudíž počet studentů, který přednášku navštěvuje, je roven počtu zapsaných studentů v předmětu. Toto ale nemusí platit v některých ročnících, kde je studentů moc na to, aby bylo v možnostech FIT VUT je všechny vměstnat na jednu přednášku. Cvičení a laboratoře se od přednášek liší tím, že se na ně studenti musí registrovat v aplikaci StudIS, v modulu Registrace vyučování. V neposlední řadě je třeba si uvědomit, že pokud se stane, že je v den plánované výuky státní svátek nebo rektorský či děkanský den, pak lze tuto neuskutečněnou výuku vykázat.

Další skupina činností se týká projektů. V tomto kontextu lze projekty chápat jako domácí úkoly zadané studentům v rámci výuky předmětu, které chce vyučující následně ohodnotit. Činnosti spojené s hodnocením projektů lze vykázat jen pokud byl projekt odevzdán nebo prezentován. V IS VUT je prvním krokem k vytvoření projektu v rámci předmětu vytvoření hodnocení v aplikaci Teacher v modulu Struktura hodnocení. K hodnocení

Specifikace aktivity			ZH
Činnost	Typ studia	Popis	
Organizační činnost	Bc. a Mgr. studium	Supergarant	10
	Bc., Mgr. a Dr. studium	Garance předmětu	$10 + 0,1 * S$
Vědecké práce studentů	Bc. a Mgr. studium	Vedení práce	$10 * P$
		Komise pro prezentaci studentských prací	$0,65 * S$
Činnost školitele doktorandů	Dr. studium	Vedení (1. až 4. rok)	$75 * S$
		Vedení (5. rok a dál)	0
		Prémie za úspěšnou disertaci doktoranda	$150 * S$

Tabulka 3.3: Tabulka popisující činnosti spojené s organizací předmětů, vědeckými pracemi studentů a činnostmi školitele doktorandů a výpočty započitatelných hodin (ZH) s nimi spojenými, kde P je počet vědeckých prací a S je počet studentů.

projektu je možné následně přidat zadání nebo termín. Pokud je k hodnocení v aplikaci Teacher v modulu Vypisování zadání přidáno zadání, tak v rámci něj studenti mohou v aplikaci StudIS projekt odevzdat a vyučující pak v aplikaci Teacher mohou projekt v modulu Hodnocení studenta ohodnotit. V rámci vypisování zadání se specifikuje, zda zadání bude hodnoceno automaticky nebo individuálně. Alternativně lze v aplikaci Teacher k hodnocení navázat v modulu Vypisování termínů termín a v rámci tohoto termínu uspořádat obhajobu projektu, která bude následně v modulu Hodnocení studenta ohodnocena. Bohužel k jednomu hodnocení nelze navázat zadání a termín zároveň. Tudíž pokud vyučující vyžaduje jak odevzdání, tak i obhajobu projektu, pak je nucen vytvořit dvě hodnocení, kde jedno slouží pro odevzdání a druhé pro obhajobu.

Kruhové konzultace jsou v systému vedeny v rámci speciálního předmětu IKK¹, který slouží právě k tomu, aby se v rámci něj pořádala setkání studentů prvního ročníku s akademiky, kteří jim dají cenné rady k jejich začátku studia na vysoké škole. V aplikaci Teacher vyučující v rámci toho předmětu v modulu Editace rozvrhu vytvoří v rozvrhu vyučování trvající 1 hodinu, které má specifikovaného vyučujícího a čas v týdnu, kdy se bude konat. Následně se jednotliví studenti v modulu Registrace vyučování mohou přihlašovat k právě jednomu vyučování. Tímto v systému vzniknou skupiny, kde pak pro každou lze vykázat až tři hodinové konzultace.

V neposlední řadě je v tabulce 3.2 zmíněno vedení řízeného studia doktorandů prezenčního doktorského studia. Vykazování vedení řízeného studia slouží v předmětech doktorského studia jako alternativa pro vykazování úvazků za přednášky, o které se v málo obsazeném předmětu musí na doktorském studiu žádat. Vedení řízeného studia se vykazuje v případě, že počet započitatelných hodin za něj je nižší, jak počet hodin vykázaných za případné přednášky. Vedení řízeného studia se v IS VUT mimo úvazky neeviduje, a i ve WIS se tento úvazek vykazuje jen manuálně.

V tabulce 3.3 se popisuje výpočet započitatelných hodin za organizační činnosti. První z nich je činnost supergaranta. Supergarant je osoba zvolená proděkanem pro bakalářské nebo magisterské studium a jmenovaná kolegiem děkana na daný semestr, která je

¹<https://www.fit.vut.cz/study/course/259513>

Specifikace aktivity			ZH
Činnost	Typ studia	Popis	
Přijímací zkoušky	Bc. a Mgr. studium	Předseda komise	$0,3 * U$
		Dozor v učebně	$0,15 * U$
		Hlavní předseda	$0,3 * \frac{U}{K}$
	Dr. studium	Každý člen komise	$1 * U$

Tabulka 3.4: Tabulka popisující činnosti spojené s přijímacími zkouškami a výpočty započítatelných hodin (ZH) s nimi spojenými, kde U je počet uchazečů, K je počet komisí.

zodpovědná za koordinaci studijních aktivit tak, aby nedocházelo k přetěžování studentů. V bakalářském i v magisterském studiu může být na jeden semestr zvolen právě jeden supergarant. Práce supergaranta se v systému nijak neeviduje a ve WIS se mu pouze vykazuje úvazek, který mu náleží, do virtuálního předmětu „Aktivity pro FIT“.

Garant předmětu garantuje předmět ve zvoleném aktuálním roce a v rámci jednotlivých akademických roků se může měnit. Proto se při zakládání předmětu pro daný akademický rok v IS VUT Apollo v modulu Předměty volí jeho garant. IS VUT Apollo uživateli nabízí předvyplnění překopírováním informací o předmětu z minulého akademického roku a tím se mu snaží urychlit práci se systémem.

Vědecké práce studentů se v IS VUT ani ve WIS nijak neevidují. Pro komise k nim náležící platí to samé.

Informace o školiteli doktoranda se do systému dostává v IS VUT Apollo v menu Fakulta při vyplňování tématu studia. Obdobným způsobem se do systému dostává informace o tzv. školiteli specialistovi, který se na práci školitele podílí a jeho podíl na práci je v systému procentuálně vyjádřený. Počet započítatelných hodin, které lze za vedení doktoranda vykázat, se mění v závislosti na tom, v jakém roku studia se doktorand nachází. Rovněž platí, že počet započítatelných hodin se mezi školitelem a školitelem specialistou rozděluje podle podílu na práci školitele, který je v systému uveden. Počet započítatelných hodin za vedení doktoranda je na jednoho školitele na jeden akademický rok zastropován. Podrobnosti k tomu, jak je počet započítatelných hodin zastropován, lze dohledat ve směrnici [10].

V tabulce 3.4 jsou popsány činnosti spojené s přijímacími zkouškami. Informace o předsedovi a členovi komise u přijímacích zkoušek se do IS VUT dostává pomocí IS VUT Apollo. Osoba, která prováděla dozor v učebně, se v systému vede jako člen komise. Záznamy o těchto činnostech se zapisují v modulu Přijímačky a k nim se udává také informace o tom, k jakému typu studia se má aktivita připsat. Informace o hlavním předsedovi se do IS VUT nezadá.

Činnosti spojené se zkoušením studentů jsou seskupeny v tabulce 3.5. Z tabulky je zřejmé, že se jedná o skupinu různorodých činností. Ve skupině je zahrnuta příprava zkoušek, samotné zkoušení, dozor u zkoušky a hodnocení zkoušky. Tato různorodost se podepsala i na tom, jak různé jsou výpočty započítatelných hodin u jednotlivých činností. Některé výpočty úvazku neberou v potaz množství studentů na zkoušce, některé berou v potaz množství studentů na zkoušce a některé dokonce porovnávají poměr jednotlivých částí zkoušky (testovou část, ústní část a část vyžadující vypracování písemné odpovědi). Poměry jednotlivých částí zkoušky může vyučující nadefinovat v aplikaci Teacher v modulu Pravidla klasifikace.

Specifikace aktivity			ZH
Činnost	Typ studia	Popis	
Příprava písemné zkoušky nebo písemné části kombinované zkoušky	Bc. a Mgr. studium	Průběžný test	$1,5 + 0,01 * S$
		Půlsemestrální zkouška	$3 * \frac{B_t+B_v}{B_t+B_v+B_u} + 0,02 * S$
		Semestrální zkouška	$6 * \frac{B_t+B_v}{B_t+B_v+B_u} + 0,04 * S$
Hodnocení písemné zkoušky nebo písemné části kombinované zkoušky	Bc. a Mgr. studium	Průběžný test	$\frac{0,05*B_t+0,1*B_v}{B_t+B_v} * S$
		Půlsemestrální zkouška	$\frac{0,1*B_t+0,2*B_v}{B_t+B_v+B_u} * S$
		Semestrální zkouška	$\frac{0,2*B_t+0,4*B_v}{B_t+B_v+B_u} * S$
Ústní zkouška	Bc. a Mgr. studium	Půlsemestrální zkouška	$0,3 * S$
		Semestrální zkouška	$0,6 * S$
Ústní část kombinované zkoušky	Bc. a Mgr. studium	Půlsemestrální zkouška	$0,3 * \frac{B_u}{B_t+B_v+B_u} * S$
		Semestrální zkouška	$0,6 * \frac{B_u}{B_t+B_v+B_u} * S$
Zkoušení doktoranda	Dr. studium	Semestrální zkouška	$1 * S$
Dozor u zkoušky	Bc. a Mgr. studium	Průběžný test	1
		Půlsemestrální zkouška	1,5
		Semestrální zkouška	2,5

Tabulka 3.5: Tabulka popisující činnosti spojené se zkoušením studentů a výpočty započítatelných hodin (ZH) s nimi spojenými, kde S je počet studentů, H je počet hodin, B_t je počet bodů za testovou část, B_v je počet bodů za část vyžadující vypracování písemné odpovědi, B_u je počet bodů za ústní část.

Specifikace aktivity			ZH
Činnost	Typ studia	Popis	
Státní závěrečné zkoušky	Bc. studium	Člen komise	$1 * S$
		Předseda komise	$1,5 * S$
	Mgr. studium	Člen komise	$2 * S$
		Předseda komise	$3 * S$
	Dr. studium	Člen komise	$3 * S$
		Předseda komise	$4,5 * S$
Obhajoba prací a projektů	Bc. studium	Semestrální práce (člen komise)	$0,25 * S$
	Mgr. studium	Semestrální práce (člen komise)	$0,33 * S$
	Bc. a Mgr. studium	Projektová praxe (člen komise)	$0,56 * S$
		Organizátor projektové praxe	$1 * S$
	Dr. studium	Disertační práce (člen komise)	$3 * S$

Tabulka 3.6: Tabulka popisující činnosti spojené se závěrečnou zkouškou a obhajobou prací a projektů a výpočty započítatelných hodin (ZH) s nimi spojenými, kde S je počet studentů a zkratky v závorkách jsou zkratkami předmětů na FIT VUT.

Když chce vyučující do systému vložit novou zkoušku, tak v první řadě musí v aplikaci Teacher k předmětu v modulu Struktura hodnocení vytvořit nové hodnocení. K tomuto hodnocení poté musí vytvořit v modulu Vypisování termínů termíny, na které buď automaticky přihlásí studenty nebo je nechá se v aplikaci StudIS na termín zkoušky zaregistrovat manuálně v modulu Registrace termínů. Informace o přípravě písemné zkoušky nebo písemné části kombinované zkoušky se do systému neukládá. Taktéž informace o tom, kdo prováděl dozor u zkoušky, bohužel nelze uložit do databázových struktur IS VUT. Posledním krokem v rámci zkoušení je zapsání výsledků do systému v aplikaci Teacher v modulu Hodnocení studenta.

V tabulce 3.6 jsou seskupeny činnosti spojené se státními závěrečnými zkouškami a obhajováním prací nebo projektů. V tabulce 3.7 jsou popsány činnosti, které jsou spojeny s vedením a oponováním prací nebo projektů. Protože ohodnocení těchto činností je vázáno pouze na počet studentů, pro výpočet úvazků spojených s těmito činnostmi je zapotřebí z databáze získat pouze informaci o tom, kdo činnost provedl a kolikrát.

Informace o projektové praxi se do systému dostávají následovně. V aplikaci Teacher v modulu Projektová praxe vyučující zadá do systému nové zadání projektové praxe. V aplikaci StudIS si student následně může zadání zobrazit a přihlásit se na něj. Komise k projektové praxi vyučující spravuje v aplikaci Teacher v modulu Semestrální práce.

Semestrální práce, bakalářská práce, diplomová práce a disertační práce spadají pod agendu závěrečných prací. Zadání závěrečné práce může vyučující vytvořit v aplikaci Teacher v modulu Vypisování zadání. V tomtéž modulu se schvaluje, zda se konkrétní zadání práce bude vypracovávat. Student se na závěrečnou práci může přihlásit pomocí aplikace StudIS v modulu Registrace závěrečných prací. Na FIT se s bakalářskou a diplomovou prací spojují práce semestrální. K semestrálním pracím vyučující může vypsát komisi v aplikaci Teacher v modulu Semestrální práce. Student si tuto komisi může zobrazit v aplikaci StudIS v modulu Moje komise. Pomocí aplikace Teacher v modulu Semestrální práce může pověřená osoba po otevření detailu komise k semestrální práci zapsat doporučené hodnocení práce studenta. Vedoucí může zapsat konečné hodnocení pomocí modulu Vypisování Zadání a dále může v tomto modulu studentovi udělit za předmět spojený se semestrální prací zápočet. Hotovou závěrečnou práci student odevzdává pomocí aplikace StudIS v modulu

Specifikace aktivity			ZH
Činnost	Typ studia	Popis	
Vedení prací a projektů	Bc. studium	Projektová praxe	5 * S
		Semestrální práce (ISP)	4 * S
		Semestrální práce (ITT)	7 * S
		Bakalářská práce (IBP)	18 * S
		Bakalářská práce (IBT)	22 * S
	Mgr. studium	Projektová praxe	5 * S
		Semestrální práce	11 * S
		Diplomová práce	29 * S
	Mobilitní programy	Projektová praxe	18 * S
		Semestrální práce	11 * S
		Bakalářská práce	33 * S
		Semestrální práce	17 * S
Diplomová práce		43 * S	
Vypracování oponentního posudku	Bc. studium	Bakalářská práce	10 * S
	Mgr. studium	Diplomová práce	12 * S
	Dr. studium	Pojednání ke státní doktorské zkoušce	10 * S
		Disertační práce	25 * S

Tabulka 3.7: Tabulka popisující činnosti spojené s vedením a oponentováním závěrečných prací a výpočty započítatelných hodin (ZH) s nimi spojenými, kde S je počet studentů a zkratky v závorkách jsou zkratkami předmětů na FIT VUT.

Moje závěrečná práce. Oponent a vedoucí následně mohou do systému zadat hodnocení závěrečné práce v aplikaci Teacher v modulu Hodnocení prací. Vypisování komisí pro státní závěrečné zkoušky a zadání finálního ohodnocení závěrečné práce do systému se provádí pomocí modulů v IS VUT Apollo, které jsou dostupné přes menu Fakulta. Konkrétně se jedná o modul „Komise pro závěrečné práce“ pro vytváření komisí a modul „Závěrečná zkouška“ pro zaznamenání hodnocení. Student může svou komisi pro státní závěrečnou zkoušku vidět opět v aplikaci StudIS v modulu Moje komise.

3.5 Omezení pedagogických úvazků

Ve směrnici ([10]) je specifikováno hned několik omezení úvazků. Ta omezení, která je možné využít pro kontrolu vykázaných úvazků, budou zmíněna v této podkapitole. Pořadí, ve kterém se budou omezení jednotlivých činností popisovat, odpovídá pořadí z podkapitoly 3.4.

První skupina omezení je spojená s tabulkou 3.2 a je shrnuta v tabulce 3.8. Prvním omezením je, že mohou být vykázány činnosti spojené s projekty těch studentů, kteří je odevzdali nebo prezentovali. Druhé omezení souvisí s kruhovými konzultacemi. U kruhové konzultace je totiž důležité, aby se pro jednu skupinu vykázaly nanejvýš tři konzultace během jednoho semestru. Poslední dvě omezení se vážou na činnost vedení řízeného studia. Je zde vhodné kontrolovat, že vykázané započítatelné hodiny za vedení řízeného studia v předmětu nepřekračují ty, které by mohly být vykázány za běžnou kontaktní výuku (přednášky, cvičení apod.). Také je zde důležité kontrolovat, že v předmětu nejsou zároveň vykázány činnosti spojené s vedením řízeného studia a běžná kontaktní výuka, tedy že práce není vykázána 2x.

Činnost	Omezení
Hodnocení a obhajoby projektů	Lze vykazovat činnost spojenou pouze s těmi studenty, kteří projekt odevzdali nebo obhajovali.
Kruhové konzultace	Lze vykázat pouze 3 konzultace na jednu skupinu v průběhu jednoho semestru.
Vedení řízeného studia	V rámci tohoto předmětu by neměla být vykázána tato činnost a zároveň činnost za přednášky, cvičení a další výukové aktivity v doktorském studiu.
Vedení řízeného studia	Počet vykázaných započítatelných hodin by neměl převyšovat počet započítatelných hodin, který by mohl být v daném předmětu vykázán za běžnou výuku (přednášky, cvičení apod.).

Tabulka 3.8: Tabulka popisující omezení spojená s činnostmi, které jsou zmíněny v tabulce 3.2.

Omezení související s tabulkou 3.3 jsou shrnuta v tabulce 3.9. První omezení zmíněné v tabulce 3.9 bude problémové na implementaci, protože úvazky se s nejvyšší pravděpodobností budou vázat pouze na vyučujícího. Z tohoto důvodu nejspíš nebude možné kontrolovat, že je úvazek vykázán nanejvýš čtvrtým rokem, protože v systému neexistuje vazba mezi vyučujícím, úvazkem a studentem. Na druhou stranu by nemělo být velkým problémem toto omezení kontrolovat v rámci jednoho akademického roku, jelikož v systému je uložena informace o tom, kdy student studium započal. Zbývá dvě omezení by taktéž neměl být problém implementovat, jelikož jediná znalost, která je pro tento výpočet potřeba s výjimkou pravidel pro krácení, je počet vykázaných započítatelných hodin za danou činnost.

První tři omezení popsaná v tabulce 3.10 limitují počet vykázaných činností stejného typu v jednom předmětu. Čtvrtý řádek tabulky omezuje vykázání půlsestrální zkoušky a průběžného testu v rámci jednoho předmětu na základě jeho způsobu ukončení. Pátý a šestý řádek tabulky podmiňuje možnost vykázání zkoušení studenta tím, že student se ke zkoušení musí dostavit. Poslední omezení zakazuje vykázání dozoru při ústním zkoušení.

Činnost	Omezení
Činnost školitele doktorandů	Činnost školitele doktoranda se vykazuje pouze pro první 4 roky studia doktoranda.
Činnost školitele doktorandů (školitel)	Počet vykázaných započítatelných hodin se krátí podle následujících pravidel. Pásmo 0-500 započítatelných hodin se násobí konstantou 1. Pásmo 500-750 se násobí konstantou 0,5. Pásmo 750-1000 se násobí konstantou 0,25. Pásmo 1000-1250 se násobí konstantou 0,125. Započítatelné hodiny v pásmu nad 1250 se násobí konstantou 0.
Činnost školitele doktorandů (školitel specialista)	Počet vykázaných započítatelných hodin se krátí podle následujících pravidel. Pásmo 0-250 započítatelných hodin se násobí konstantou 1. Pásmo 250-375 se násobí konstantou 0,5. Pásmo 375-500 se násobí konstantou 0,25. Pásmo 500-625 se násobí konstantou 0,125. Započítatelné hodiny v pásmu nad 625 se násobí konstantou 0.

Tabulka 3.9: Tabulka popisující omezení spojená s činnostmi, které jsou zmíněny v tabulce 3.3.

Činnost	Omezení
Činnosti spojené s půlsestrální zkouškou	Pro předmět zakončený zkouškou lze vykázat maximálně 1 termín půlsestrální zkoušky. Pro předmět zakončený zápočtem, klasifikovaným zápočtem nebo kolokviem lze vykázat maximálně 2 termíny půlsestrální zkoušky. Termín se může skládat ze 2 běhů na 1 přednáškovou skupinu.
Činnosti spojené s průběžným testem	Pro předmět zakončený zkouškou lze vykázat maximálně 2 termíny průběžného testu. Pro předmět zakončený zápočtem, klasifikovaným zápočtem nebo kolokviem lze vykázat maximálně 4 termíny průběžného testu. Termín se může skládat ze 2 běhů na 1 přednáškovou skupinu.
Příprava písemné zkoušky nebo písemné části kombinované zkoušky	Přípravu lze vykázat maximálně pro 5 termínů zkoušky.
Činnosti spojené s půlsestrální zkouškou	V rámci jednoho předmětu zakončeného zkouškou nelze půlsestrální zkoušku vykázat společně s průběžným testem.
Hodnocení písemné zkoušky nebo písemné části kombinované zkoušky	Úvazek lze vykázat pouze za studenty, kteří se ke zkoušce dostavili.
Zkoušení doktoranda	Úvazek lze vykázat pouze za studenty, kteří se ke zkoušce dostavili.
Ústní zkouška	Pro ústní zkoušku bez písemné části nelze vykázat dozor.

Tabulka 3.10: Tabulka popisující omezení spojená s činnostmi, které jsou zmíněny v tabulce 3.5.

3.6 Analýza akcí prováděných s pedagogickými úvazky

Do systému je nutné vkládat nové úvazky a spravovat ty stávající. Úvazek se vždy váže na konkrétní předmět v daném semestru. Z tohoto důvodu by bylo vhodné filtrovat úvazky a přístup k nim na základě předmětu, školního roku a semestru. V rámci zadání je specifikováno, že se mají činnosti spojené s pedagogickými úvazky co nejvíce automatizovat. Při vykazování úvazků mohou ovšem nastat situace, kdy informace v IS nemusí přesně odpovídat tomu, co se ve skutečnosti stalo. Například písemky se v systému mohou jevit jako opravené jednou osobou, i když ve skutečnosti se mohlo jednat o spolupráci několika osob, kde jedna měla za úkol zapsat výsledky do IS. Z tohoto důvodu je důležité, aby úvazek před zadáním do IS zkontroloval garant předmětu, protože za vykázané úvazky zodpovídá. Možným způsobem automatizace procesu vykazování úvazků by tedy bylo vytvoření úvazků z dat dostupných v IS a nabídnutí úvazků uživateli, který je může zkontrolovat a uložit jako validní úvazky.

Pro přehlednost při schvalování navržených úvazků a při kontrole by podle mě bylo vhodné uživateli zobrazovat uložené úvazky v seznamu. Kontrolování vykázaných úvazků podle mě urychlí zobrazení přehledu chyb a nesrovnalostí, které systém ve vykázaných úvazcích najde, a možnost zkontrolovat úvazek již v zobrazeném seznamu bez nutnosti otevírání detailu každého správně vykázaného úvazku.

Ze zadání práce vyplývá ještě nutnost zobrazení osobního přehledu úvazků a přehledu úvazků pro vedení fakulty a vědecké oddělení. Na základě diskuzí s vedoucím mé práce a paní referentkou pro vědu a výzkum na FIT jsem došel k následujícím závěrům. Osobní přehled úvazků je seznam vykázaných úvazků uživatele, které jsou vázány na něj, a je možné je filtrovat podle akademického roku. Přehled úvazků pro vedení fakulty a vědecké oddělení je seznam osob, který lze filtrovat podle akademického roku a zařazení osob k organizačním jednotkám FIT. V seznamu je pak zapotřebí u osoby zobrazit kromě jejího jména i kolik započítatelných hodin vykážala v daném akademickém roce a v případě, že se jedná o doktoranda, tak zobrazit, kolik započítatelných hodin měl v daném akademickém roce vykážat.

Jak již bylo zmíněno, započítatelné hodiny se používají pro výpočet platů. Tuto problematiku jsem konzultoval s děkanem fakulty. Z konzultace vyplynulo, že je zapotřebí vytvořit export vykázaných započítatelných hodin na osobu v rámci jednoho semestru. Dále je zapotřebí zobrazit sumu vykázaných započítatelných hodin pro jednotlivé ústavy za jeden semestr. Posledním požadavkem byla možnost u osoby zaznamenat, jaké množství započítatelných hodin jí má být proplaceno, a kolik hodin má být posunuto ústavu, pod kterým pracuje.

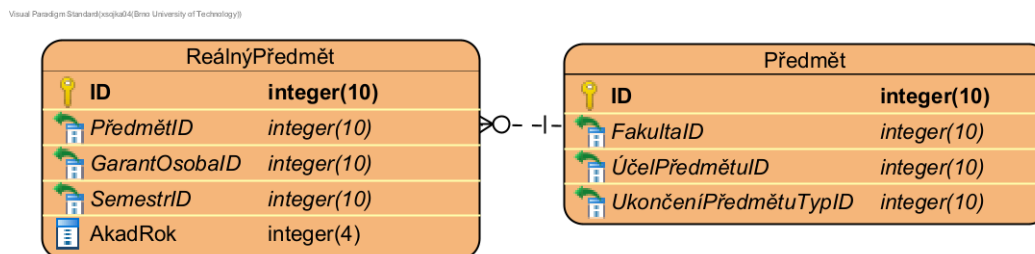
3.7 Analýza základních databázových schémat a entit

Struktury v databázi jsou rozčleněny do několika desítek hlavních schémat. Pro většinu této práce budu potřebovat dvě z těchto schémat. Schéma, pomocí kterého se spravují informace o osobách evidovaných v systému, a schéma, které obsahuje agendu spojenou s výukou. Ve zbytku této podkapitoly je popsáno, jak se budou prezentovat v rámci této práce datové struktury a základní informace o zmíněných schématech.

Kolegové vyvíjející IS VUT mě požádali, abych v práci nezveřejňoval názvy schémat, tabulek a sloupců. Z tohoto důvodu nebudu v práci používat existující názvy a používané názvy budou mnou vytvořené. Budu se snažit vytvářet takové názvy tabulek a sloupců, které budou výstižně popisovat jejich význam.

Schéma obsahující osoby evidované v IS VUT obsahuje pouze několik tabulek, které je zapotřebí pro účely této práce vysvětlit. Základní záznamy o osobách jsou seskupeny v tabulce Osoba. Druhou důležitou tabulkou pro potřeby této práce je tabulka držící záznamy o univerzitních celcích. Univerzitní celek reprezentuje nějakou část univerzity (například děkanát, ústav nebo oddělení). Tyto celky jsou pak pomocí další tabulky seskupeny do stromové struktury. Jednotlivé osoby evidované v systému následně mohou být pomocí vazebních tabulek navázány na univerzitní celek, společně s rolí, kterou v rámci celku zastávají.

Ve schématu popisujícím agendu výuky se spravuje většina informací nutných pro generování a kontrolu pedagogických úvazků. Jak by se dalo v této části databáze očekávat, mnoho věcí se zde váže na akademický rok. Akademický rok není v databázi entitou. V databázi se používá pouze jeho číselná reprezentace jako identifikátor. Frekventovaně odkazovanou entitou v tomto schématu databáze je Předmět a ReálnýPředmět. Entita Předmět reprezentuje konkrétní předmět, který je vyučován. Entita ReálnýPředmět reprezentuje předmět vázaný na konkrétní semestr, ve kterém je vyučován.



Obrázek 3.2: Entitně vztahový model popisující způsob uložení informací o předmětu v databázi IS VUT.

V diagramu 3.2 je znázorněno, jak se entita ReálnýPředmět váže na entitu Předmět. Dále lze z diagramu vyčíst, že předmět se váže na fakultu, která jej vyučuje. Předmět má také vazbu na účel předmětu, pomocí kterého lze získat informaci o tom, zda má předmět nějaký speciální účel (například závěrečná zkouška nebo projektová praxe). Poslední zde zmíněnou vazbou předmětu je vazba na typ ukončení předmětu, pomocí kterého lze například zjistit, zda se předmět ukončuje zápočtem nebo zkouškou. Jak již bylo zmíněno výše, reálný předmět má vazbu na předmět, semestr a akademický rok. Dále je zde vazba na osobu garanta, který se může v průběhu let v rámci jednoho předmětu měnit.

Kapitola 4

Technologie používané pro vývoj IS VUT

V této kapitole jsou popsány technologie, které se používají pro vývoj IS VUT, který je popsán v kapitole 2.

4.1 Delphi

Delphi je vývojové prostředí vyvinuté firmou Borland pro vývoj systémů ve stejnojmenném programovacím jazyce. Jazyk Delphi je objektově orientovaným rozšířením jazyka Pascal a přidává do něj podporu pro obsluhování výjimek, tvorbu dědičnosti nebo například implementaci rozhraní. Vývojové prostředí Delphi nabízí rychlý vývoj uživatelského prostředí, kde uživatel může jednotlivé komponenty umisťovat na stránku pomocí techniky drag-and-drop. Komponenty pro tvorbu uživatelského prostředí Delphi nabízí z vizuální knihovny komponent (VCL) [11]. Pro vývoj IS VUT Apollo se aktuálně používá Delphi ve verzi 11.2.

4.2 PHP

PHP je programovací jazyk, který lze použít pro tvorbu webových stránek, které jsou vykreslovány na straně serveru. Interpret PHP je otevřený software a je multiplatformní. Z tohoto důvodu jej lze spustit na více operačních systémech (například Microsoft Windows, macOS, Linux) a webových serverech (například Apache, IIS, nginx). PHP nabízí programátorovi možnosti jak pro procedurální, tak pro objektově orientované programování. V roce 2023 běží IS VUT Web na PHP ve verzi 7.4. Systém se ještě nepřesunul na PHP ve verzi 8, protože servery, na kterých je webová aplikace provozována, mají nainstalovaný zpoplatněný operační systém Oracle Linux ve verzi 8, na kterém je PHP 7.4 zaručena podpora až do roku 2029 [3].

PHP je dynamicky typovaný programovací jazyk se slabou typovou kontrolou. Dle mého názoru tyto vlastnosti dokážou velmi urychlit vývoj nových aplikací, ale zároveň mohou být přítěží v dlouhodobě vyvíjeném produktu. V takovém produktu se totiž některé sekce kódu mohou v průběhu životního cyklu produktu přepisovat hned několikrát a může se stát, že při implementaci nových postupů vzniknou typové chyby, které by v jiném případě mohla odhalit statická typová kontrola [1].

4.3 Framework Zend

Zend je framework pro tvorbu webových stránek pomocí jazyka PHP. Nástroj implementuje architekturu MVC, nabízí komponenty pro opětovné použití a mnoho dalších funkcionalit za účelem urychlení vývoje produktu a zvýšení jeho dlouhodobé udržitelnosti [2].

Aplikace Teacher je momentálně implementována pomocí frameworku Zend ve verzi 1.11.5. Aplikace běží na verzi 1, jelikož nároky na implementační úpravy nutné pro přechod do verze 2 byly příliš velké. Z tohoto důvodu bylo rozhodnuto o nepřecházení na novou verzi a udržování produktu na verzi 1.11.5. Údržba takového produktu zahrnuje i údržbu frameworku Zend, ve kterém se s postupem času objevovaly chyby.

4.4 Nette

Nette je framework pro tvorbu webových aplikací pomocí jazyka PHP s otevřeným zdrojovým kódem. V rámci Nette se pro ladění využívá nástroj Tracy a šablonovací nástroj Latte. Hlavním vývojářem Nette a jeho původním autorem je český vývojář David Grudl a i díky tomu je podstatná část dokumentace k Nette dostupná českým jazykem [7]. Pro vývoj IS VUT Web se aktuálně používá Nette ve verzi 3.1. Verze 4.0 není nasazená z toho důvodu, že vyžaduje PHP 8.0 a novější.

Při tvorbě aplikace se v Nette využívá architektury MVP (Model-View-Presenter), která se od MVC (Model-View-Controller) liší tím, že pohled komunikuje s presentérem/kontrolerem pomocí rozhraní. Využití rozhraní si klade za cíl oddělení logiky od částí, která se stará o zobrazování [8].

4.5 SQL

SQL je standardizovaný strukturovaný dotazovací jazyk pro práci s relačními databázemi. Pomocí SQL lze definovat databázové struktury, získat data z databáze, modifikovat data, modifikovat přístup k datům nebo definovat integritní omezení. První oficiální standard pro SQL byl vydán v roce 1986 standardizačními organizacemi American National Standards Institute (ANSI) a International Standards Organization (ISO). Standard byl od té doby několikrát rozšířen o nové funkcionality. Poslední rozšíření standardu bylo vydáno v roce 2019¹ [6].

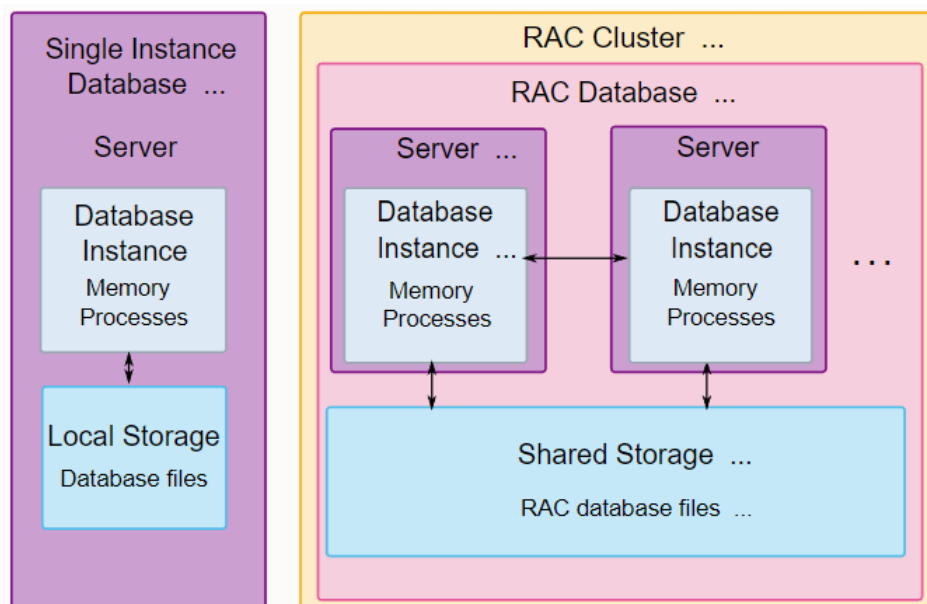
4.6 Oracle Database

Oracle database je systém řízení báze dat, který podporuje relační i objektové databáze a byl vyvinut firmou Oracle. Pro IS VUT se aktuálně používá Oracle Database ve verzi 19c. Pro práci s databázovým systémem se využívá dotazovací jazyk Oracle SQL, který rozšiřuje standardizovaný jazyk SQL. Na serverové straně lze využít jeho procedurální nadstavbu PL/SQL anebo jazyk Java [4].

Na schématu 4.1 je zobrazeno porovnání 2 způsobů, jak může být Oracle Database nakonfigurována. Vlevo je zobrazeno schéma databáze, která na serveru běží pouze na jedné instanci. Je zde zobrazeno, že na serveru běží databázová instance, v rámci které, se pracuje s pamětí, a také že na serveru jsou uložena data databáze. Na schématu vpravo je ukázka nástroje Oracle Real Application Clusters (RAC), který umožňuje rozložit zátěž na

¹<https://www.iso.org/standard/67382.html>

databázi, a tak zvýšit její dostupnost. Na schématu jsou zobrazeny servery, kde na každém běží instance databáze, která pracuje s daty. Podle [5] může mít databáze RAC až 100 takovýchto serverů. Tyto servery mají sdílené úložiště, které obsahuje data databáze. Tyto servery a sdílené úložiště tvoří databázi RAC. RAC Cluster (shluk) pak následně tvoří jedna nebo více takovýchto databází [5].



Obrázek 4.1: Schéma architektury Oracle Real Applications Clusters. Převzato z [5].

Kapitola 5

Návrh

Následující kapitola rozebírá problematiku návrhu jednotlivých komponent systému, které je nutné vytvořit, pro moduly pro vykazování a kontroly pedagogických úvazků. Jak jsem se již zmínil v 3.7, při popisu databáze v entitně vztahových diagramech (ERD) na žádost kolegů vyvíjejících IS VUT nepoužívám názvy tabulek a sloupců z databáze a namísto nich používám názvy mnou vytvořené. Pro zlepšení přehlednosti diagramů jsem se rozhodl z tabulek odstranit sloupce, které pro řešení diplomové práce nebyly podstatné. Toto se týká zejména sloupců systémových.

Ve zbytku této kapitoly jsou popsány případy užití spojené s vytvářenými moduly. Následuje návrh ukládání a získávání dat o pedagogických úvazcích v rámci IS VUT a uživatelského rozhraní vytvářených modulů.

5.1 Návrh diagramu případů užití

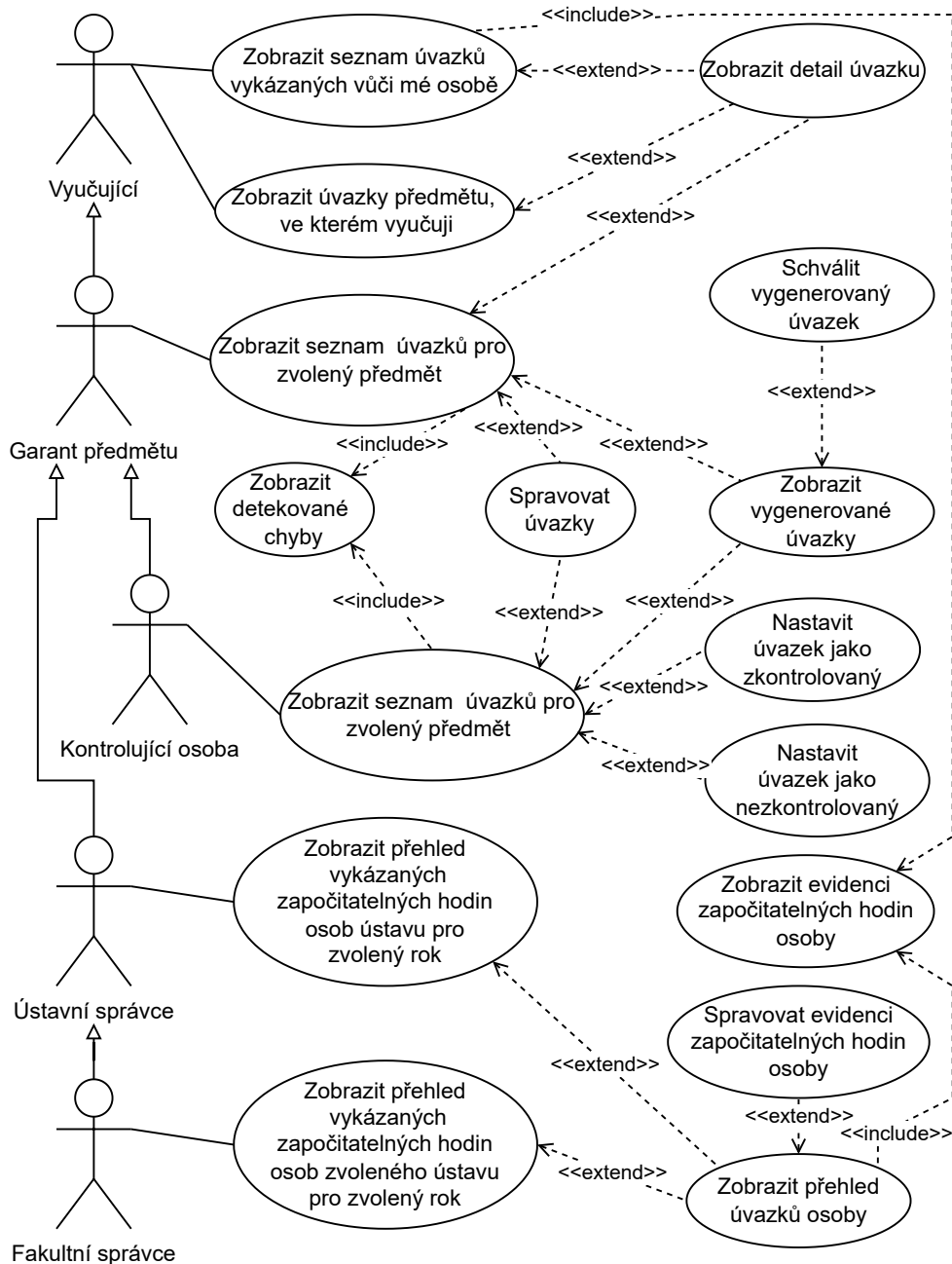
V diagramu 5.1 jsou popsány jednotlivé případy užití, které by měly být implementovány v rámci této práce. Vyučující v tomto diagramu reprezentuje osobu, která je nějakým způsobem asociována s výukou na fakultě. Taková osoba bude moci na základě akademického roku zobrazit vykázané úvazky, které jsou s ní spojovány, a evidenci započítatelných hodin, která se na ni váže. Taktéž si bude moci zobrazit úvazky v předmětech, ve kterých vyučuje. Osobou garanta předmětu může být pro účely tohoto diagramu případně i správce nebo koordinátor předmětu, který mu s vykazováním úvazků pomáhá. Možnosti garanta předmětu jsou oproti vyučujícímu rozšířeny o schopnost spravovat úvazky v rámci předmětu. Možnosti kontrolující osoby jsou oproti garantovi rozšířeny o možnost nastavení příznaku u úvazku, zda byl a nebo nebyl zkontrolován.

Ústavní správce má možnost zobrazit si pro zvolený akademický rok seznam osob asociovaných s ústavem, počet na ně vykázaných započítatelných hodin a možnost spravovat evidenci započítatelných hodin osoby.

5.2 Návrh databázové struktury pro ukládání pedagogických úvazků

Kvůli tomu, že v IS VUT se doposud informace o započítatelných hodinách neuchovávaly, je zapotřebí navrhnout a implementovat databázovou strukturu, která tyto informace bude schopna uchovávat.

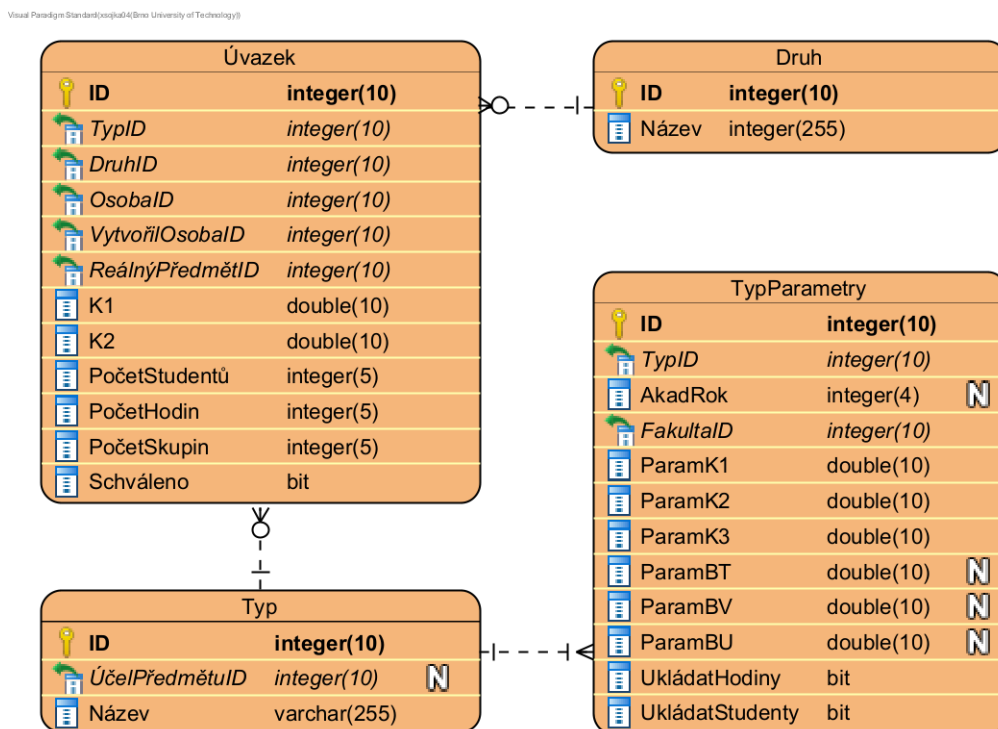
V diagramu 5.2 je znázorněn návrh, jak by se data spojená s pedagogickými úvazky měla v databázi uchovávat. Entita Úvazek uchovává informaci o tom, kdo činnost spojenou s úvazkem vykonal a kdo ji do systému zadal, a ke kterému předmětu se činnost váže. Problematika spojená s entitou Předmět a ReálnýPředmět je popsána v podkapitole 3.7.



Obrázek 5.1: Návrh diagramu případů užití pro případy užití spojené se správou pedagogických úvazků.

Dalším atributem entity Úvazek je atribut Schváleno, který slouží k uchování informace o tom, zda byl úvazek zkontrolován pověřenou osobou. Atributy K1, K2, PočetStudentů a PočetHodin jsou proměnné z rovnice 3.1. V neposlední řadě je zde atribut PočetSkupin,

kterým se celá rovnice 3.1 násobí. Tento atribut má uživateli zjednodušit správu úvazků v momentě, kdy se na jednoho vyučujícího vykazuje vícero stejných úvazků.



Obrázek 5.2: ERD popisující návrh způsobu uložení informací nutných pro správu započítatelných hodin v IS VUT.

Entita Úvazek se váže na entitu Druh. Druh je entita, která slouží jako číselník, který popisuje, jak se informace o úvazku do systému dostala a jak je zamýšlena. Bude moci nabývat následujících hodnot: „plán“, „skutečnost“ a „speciální“.

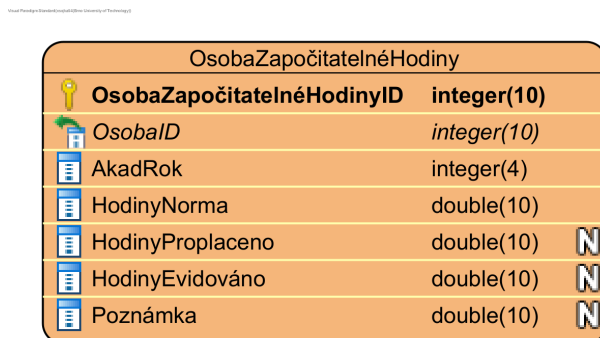
Entita Typ bude uchovávat informaci o tom, za jakou činnost byl úvazek vykázán. Atribut ÚčelPředmětID ukládá vazbu na entitu ÚčelPředmětu. Pomocí této vazby bude možné omezit využívání některých typů úvazků jen na některé předměty. Například není vhodné nabízet typ úvazku pro kruhové konzultace v předmětu, který kruhové konzultace neumožňuje vykázat. Hodnoty K1 a K2 v entitě Úvazek nejsou uživatelem zadané hodnoty, nýbrž hodnoty vypočítané z databáze. Pro to, aby bylo možné tyto konstanty v databázi vypočítat, je zapotřebí mít informace pro tento výpočet uložené. Tyto informace jsou uloženy v entitě TypParametry. Nejsou uloženy v entitě Typ kvůli tomu, že je pravděpodobné, že se tyto informace v průběhu let budou měnit, a že tyto informace bude potřeba mít zpětně dohledatelné v databázi. Entita TypParametry je limitována akademickým rokem, kdy je platná, a vazbou na fakultu, pro kterou je validní. Atributy s prefixem „Param“ slouží pro výpočet konstant. Atributy UkládatHodiny a UkládatStudenty budou pomáhat při zjednodušování formuláře pro správu úvazků.

Jak bylo zmíněno výše, tak ne každý typ úvazku bude možné použít s libovolným předmětem. Účel předmětu se musí shodovat s účelem předmětu pro daný typ úvazku a hodnota definující akademický rok reálného předmětu se musí shodovat s hodnotou akademického

roku v entitě TypParametry. Druhá možnost, kdy se může reálný předmět vázat na daný typ, je, když účel předmětu je nevyplněný a shodují se pouze v akademickém roce.

5.3 Návrh databázové struktury pro evidenci započítatelných hodin

Jak bylo zmíněno v návrhu (3.6), tak do databáze je zapotřebí ukládat informaci o tom, kolik započítatelných hodin má za povinnost doktorand odpracovat za akademický rok. Dále je zapotřebí v databázi držet informaci o tom, kolik započítatelných hodin za akademický rok má být osobě proplaceno. Z tohoto důvodu jsem navrhl tabulku 5.3, do které se budou tyto informace ukládat. V této tabulce existuje integritní omezení, které zamezuje tvorbě duplicitních záznamů. Omezení nedovoluje vytvořit 2 záznamy, které se vážou na stejnou osobu a akademický rok zároveň. Atribut HodinyNorma slouží pro ukládání informace o tom, kolik započítatelných hodin má osoba za povinnost v daném akademickém roce vykázat. Atribut HodinyProplaceno značí, kolik hodin má být osobě v daný rok proplaceno, a atribut HodinyEvidováno umožňuje zaznačit množství započítatelných hodin, které byly použity pro výpočet platů v daném akademickém roce.



Obrázek 5.3: ERD popisující způsob uložení informací spojených s evidencí započítatelných hodin.

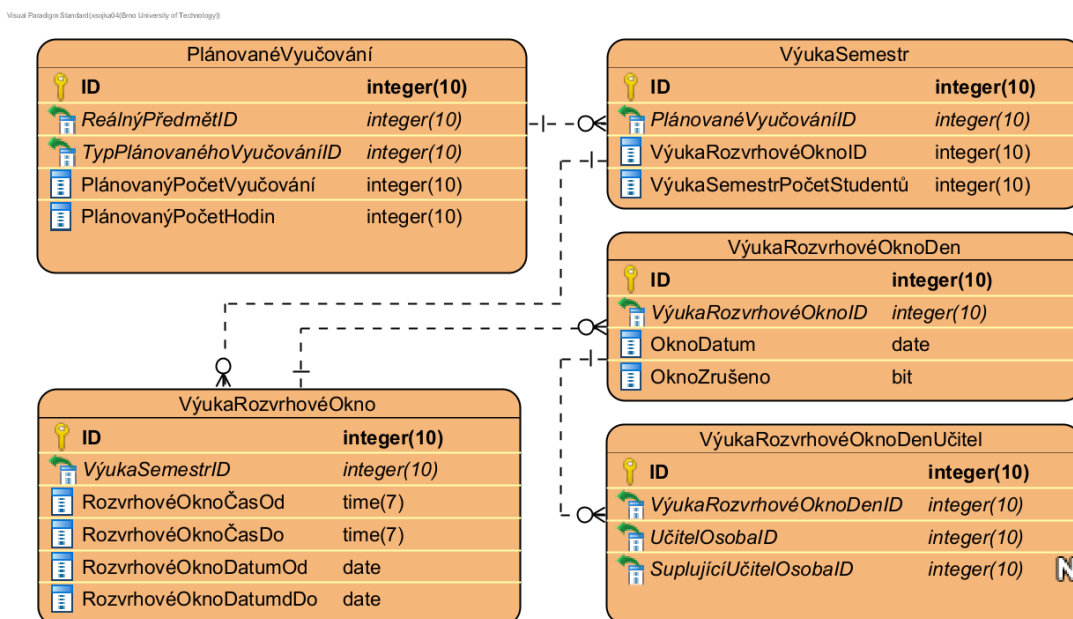
5.4 Návrh získávání dat pro generování pedagogických úvazků

Další částí nutnou pro automatizaci procesu vykazování a kontroly pedagogických úvazků je získání informací o vykonané činnosti z IS VUT a transformování těchto informací na úvazky, které budou nabídnuty uživateli ke schválení, nebo na znalosti, které mohou být použity pro kontrolu vykázaných úvazků. Ve zbytku této podkapitoly se budu věnovat tomu, jak jsou v databázi informace o činnostech popsány v tabulkách v podkapitole 3.4 uloženy.

Úvazky se budou vykazovat v rámci reálného předmětu, na který se následně budou všechny vázat. Z tohoto důvodu můžeme počítat s tím, že identifikátor entity ReálnýPředmět pro nás bude známou hodnotou a budeme ji moci použít pro vyhledávání činností.

První činnost, kterou bych chtěl popsat, je garantování předmětu z tabulky 3.3. Jak je v diagramu 3.2 zřejmé, tak získat informaci o tom, která osoba garantuje daný předmět, bude velmi jednoduché za předpokladu, že známe identifikátor reálného předmětu.

V diagramu 5.4 je popsán způsob, jakým se v databázi ukládá informace o jednotlivých oknech v rozvrhu a jak se tato informace váže na entitu ReálnýPředmět.



Obrázek 5.4: ERD popisující způsob uložení informací obsažených v rozvrhu předmětu.

Další skupina činností, o kterých lze získat z databáze IS VUT informace, se nachází v tabulce 3.2. Konkrétně se jedná o přednášky, demonstrační cvičení a cvičení a laboratoře. Informace o tom, jaký typ činnosti je v plánu v předmětu vykazovat, najdeme v entitě plánované vyučování spolu s informací o tom, kolik hodin bude mít jeden blok výuky. Další důležitá informace, kterou entita nese, je hodnota typu plánovaného vyučování. Tato hodnota se váže na číselník a na základě této hodnoty lze rozhodnout, zda se jedná o přednášku, demonstrační cvičení nebo cvičení a laboratoře. Na tuto entitu se dále váže entita VýukaSemestr. Tato entita reprezentuje už konkrétní instanci plánovaného vyučování, ke které se mohou již přiřadit konkrétní studenti. Další entitou je VýukaRozvrhovéOkno. Tato entita reprezentuje jedno souvislé okno v rozvrhu. Entita VýukaRozvrhovéOknoDen reprezentuje rozvrhové okno v konkrétní den. Uchovává tedy informaci o datu, na které se výukové okno váže, a informaci o tom, zda nebyla výuka v tomto termínu zrušena (atribut OknoZrušeno). Dále se váže na entity, které drží informaci o tom, kdo v daný den vyučuje a zda náhodou není suplován někým jiným. V neposlední řadě je zapotřebí navázat reálný předmět na typ studia, aby bylo možné rozeznat doktorské předměty od ostatních. Entita ReálnýPředmět se váže na entitu semestru, která se dále váže na entitu StudijníRočník. Studijní ročník již lze spojit s entitou Obor. V diagramu 5.6 je popsáno, jak se tato entita váže na studijní program, který již obsahuje informaci o tom, zda je předmět doktorský.

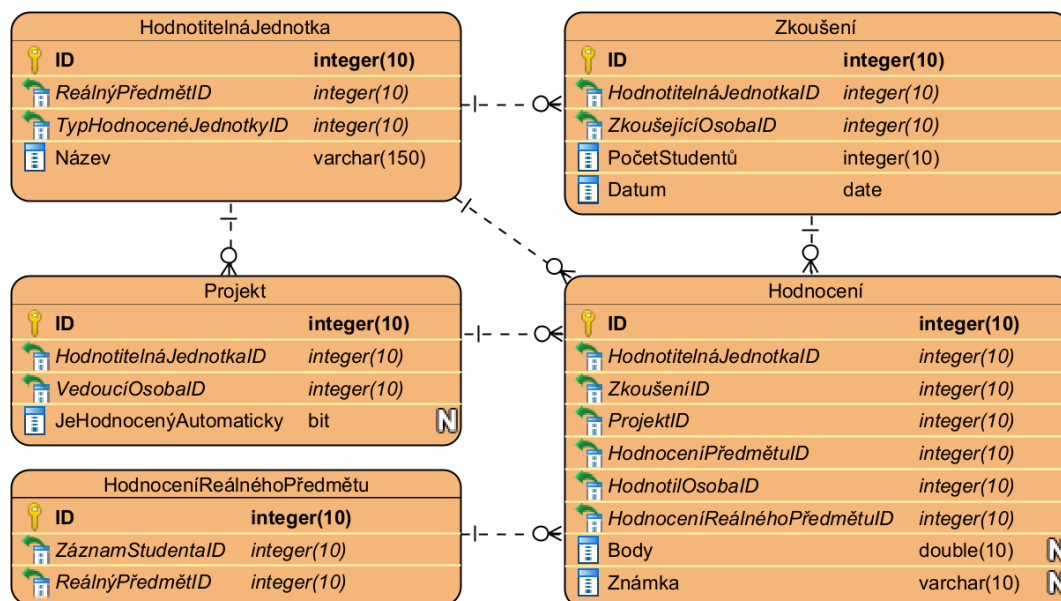
V tabulce 3.2 je zaznačeno, že pro výpočet započitatelných hodin je zapotřebí znát počet studentů a počet hodin. Další potřebnou informací je identifikátor osoby, která činnost provedla. Pro získání informací potřebných pro daný úvazek je pak zapotřebí udělat následující. Navázat entitu VýukaRozvrhovéOknoDenUčitel na daný reálný předmět. Důležité je zde navázat pouze ty záznamy, které lze navázat pomocí entity VýukaRozvrhovéOknoDen, která nemá nastavený příznak OknoZrušeno. V neposlední řadě bude zapotřebí záznamy seskupit pomocí identifikátoru entity VýukaSemestr a vyučujícího, aby bylo možné vykázat

činnosti vyučujícího v co nejmenším počtu úvazků. Dále pak bude důležité nezapomenout vykázat výuku, která byla zrušena kvůli kolizi se státním svátkem. Naštěstí v IS VUT existuje tabulka obsahující data státních svátků. Stačí tedy obdobným způsobem extrahovat záznamy zrušené výuky a navázat je na tabulku obsahující data státních svátků.

V tabulce 3.2 je ještě zmíněná činnost, která je spojená s kruhovými konzultacemi. Tyto kruhové konzultace jsou na FIT využívány pouze v rámci jednoho konkrétního předmětu. Tuto skutečnost je zapotřebí ověřit při vykazování úvazků. Jak je naznačeno na obrázku 3.2, je možné zkontrolovat, zda reálný předmět, ke kterému se snažím najít úvazky v databázi, lze navázat na předmět s identifikátorem předmětu, v rámci kterého se kruhové konzultace vykazují. Po této kontrole by mělo být možné získat záznamy o konkrétních kruhových konzultacích obdobně, jako záznamy o přednáškách. Jediným rozdílem v dotazu by měla být hodnota typu plánovaného vyučování.

Získávání informací o činnostech z tabulky 3.2, které jsou spojené s projekty, se dá popsat pomocí diagramu 5.5. Každé zkoušení (v IS VUT Web se používá slovo termín) nebo projekt (v IS VUT Web se používá slovo zadání) jsou navázány na hodnotitelnou jednotku, která sjednocuje tyto dvě entity. Hodnotitelná jednotka může mít vazbu buď na projekt, nebo na zkoušení. V systému není podporovaný stav, kdy by byla navázána zároveň na obojí. Projekt obecně představuje takový úkon, kdy zadávající očekává, že student bude muset v rámci této hodnotitelné jednotky něco odevzdat (typicky do systému). Na druhou stranu zkoušení je obecně něco, co má stanovený čas a místo, na které se musí na zkoušení zaregistrovaný student někam dostavit. Zkoušením může být písemný test, zkouška nebo například obhajoba projektu.

Visual Paradigm Standard (www.vp4.com University of Technology)



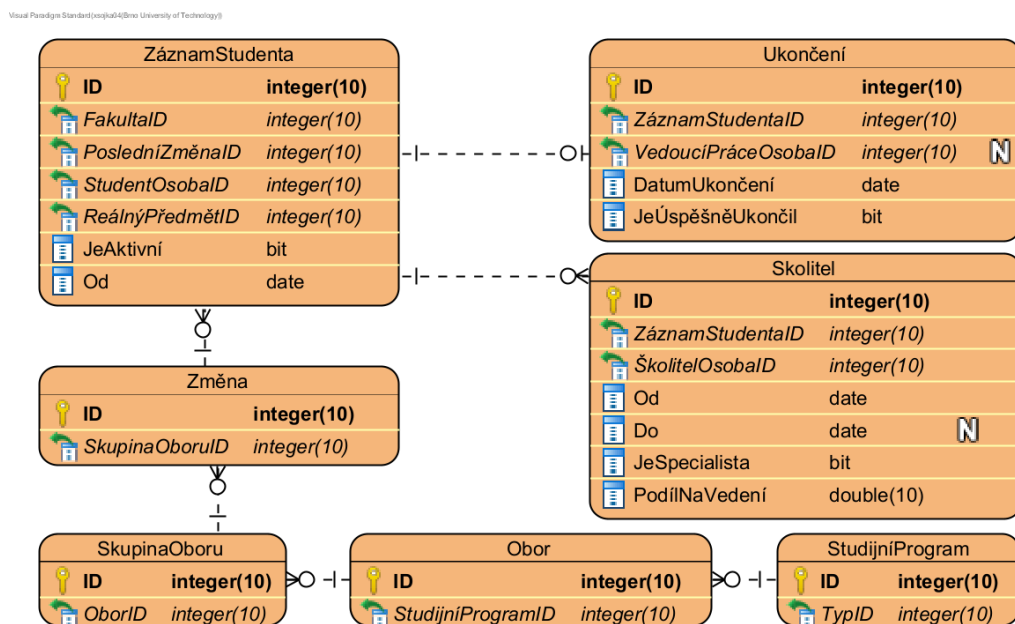
Obrázek 5.5: ERD popisující způsob uložení informací spojených s projekty a zkouškami.

Informace pro úvazky spojené s hodnocením projektu lze tedy nalézt následujícím způsobem. Nejprve je zapotřebí najít hodnotitelné jednotky, které jsou navázané na uživatelem zvolený předmět a typ hodnocené jednotky odpovídá projektu. Druhým krokem je nalézt všechna hodnocení, která jsou na tuto hodnotitelnou jednotku navázána přes projekt. In-

formace o tom, zda byl projekt hodnocen automaticky, je uložena v entitě Projekt. Osoba, které bude připsán úvazek za tuto činnost, bude specifikována atributem HodnotilOsobaID v entitě Hodnocení.

Záznamy pro tvorbu úvazku za obhajobu projektu je zapotřebí hledat přes tabulku Zkoušení. Opět je zapotřebí lokalizovat hodnotitelné jednotky, které jsou navázané na uživatelem zvolený předmět. Tentokrát jejich typ ale musí odpovídat typu pro obhajobu projektu. Dalším krokem je opět najít všechna hodnocení, která jsou ale tentokrát navázána na hodnotitelnou jednotku pomocí zkoušení. Jako osoba, které bude připsán úvazek za tuto činnost, bude tentokrát vybrána zkoušející osoba, která je uvedena u zkoušení.

Výpočty započitatelných hodin k činnostem školitele doktorandů jsou popsány v tabulce 3.3. Pro tyto výpočty je zapotřebí znát počet studentů a školitele. Datové struktury potřebné k zjištění těchto informací jsou znázorněny v diagramu 5.6.



Obrázek 5.6: ERD popisující způsob uložení informací spojených s činnostmi školitele doktorandů.

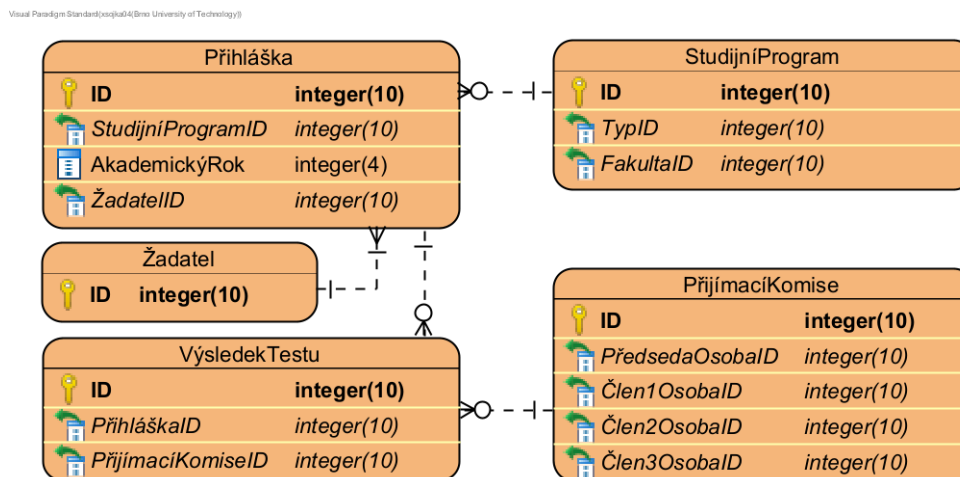
První je zapotřebí z databáze získat studenty doktorského studia. ZáznamStudenta je entita, která reprezentuje celé studium studenta, a proto se na něj i váže. Aby se z databáze vyfiltrovaly pouze záznamy studentů doktorského studia, je zapotřebí záznam studenta navázat na entitu StudijníProgram, která bude mít hodnotu typ studijního programu nastavenou na „doktorský“. Způsob navázání těchto dvou entit je popsán v diagramu 5.6.

Získat záznamy o doktorandech, kteří úspěšně dokončili studium, je již jednoduché. Stačí pouze navázat entitu ZáznamStudenta na entitu Ukončení. Entita Ukončení již pak obsahuje informaci o tom, zda ukončení studia bylo úspěšné a kdo vedl jeho disertační práci.

Získat informace o tom, jak mají být odměněni školitelé doktoranda v průběhu jeho studia, již není tak jednoduché. Opět to obnáší navázání na entitu Ukončení, která se bude moci využít k tomu, aby se vyfiltrovali doktorandi, kteří studium ukončili dříve než v analyzovaném akademickém roce. Tento akademický rok je možné získat z entity ReálnýPředmět, která je znázorněna v diagramu 5.4. Pro zjištění aktuálního roku studia k analyzovanému akademickému roku je tedy zapotřebí hodnota akademického roku z entity ReálnýPředmět,

datum začátku záznamu studenta a případně datum jeho ukončení z entity Ukončení. Dále je zapotřebí navázat záznam studenta na entity Školitelů, které obsahují informaci o tom, kdo a kdy byl školitelem doktoranda, zda byl školitelem specialistou a jaký měl podíl na vedení doktoranda v daném období.

Pro vypočtení započitatelných hodin za přijímací zkoušky je zapotřebí znát předsedu komise a počet studentů, které měla na starost. V případě přijímacího řízení na doktorské studium je zapotřebí znát všechny členy komise. Výpočet pro tyto činnosti je uveden v tabulce 3.4.



Obrázek 5.7: ERD popisující způsob uložení informací spojených s přijímacími zkouškami.

V diagramu 5.7 je ukázáno, že přihlášky lze filtrovat pomocí akademického roku, který jsme schopni získat z entity ReálnýPředmět, jejíž identifikátor budeme mít k dispozici při hledání činností. Takto získané záznamy je možné navázat na komise pomocí entity VýsledekTestu. Informace o tom, kdo byl předsedou a členem komise, jsou uloženy v entitě PřijímacíKomise. Důležitým krokem je dále zjistit, v rámci kterého typu studia fakulty byla přihláška podána. Toto se dá zjistit navázáním na entitu StudijníProgram, kde je uveden identifikátor fakulty a typ programu, na který se žadatel snaží být přijat.

V tabulce 3.5 jsou popsány činnosti spojené se zkoušením a testováním studentů v rámci předmětu. Způsob uložení těchto informací je zachycen v diagramu 5.5.

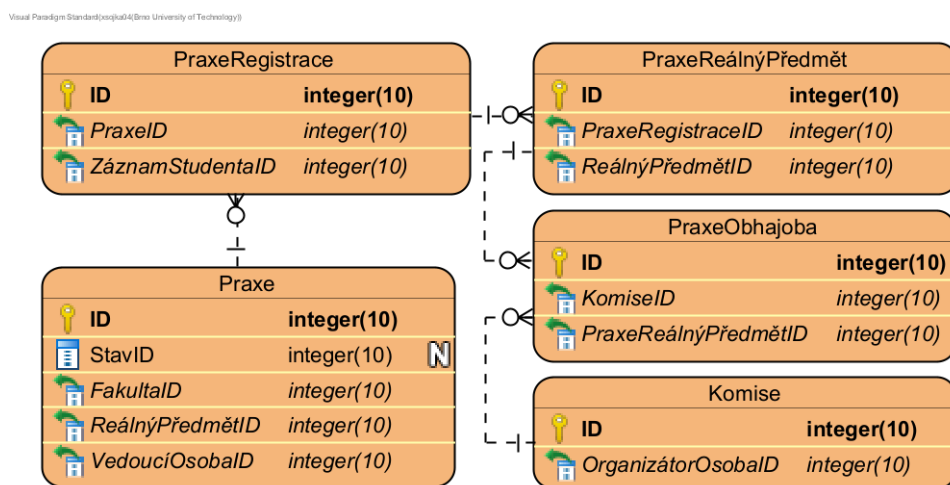
Extrakce informací o činnostech spojených s průběžnými testy a půlsestrálními zkouškami je velmi podobná. Nejprve je zapotřebí identifikovat záznamy hodnotitelných jednotek, které mají identifikátor daného reálného předmětu. Následně se mohou vyfiltrovat ty záznamy, které mají typ hodnotitelné jednotky pro průběžný test nebo pro půlsestrální zkoušku. Navázáním na entitu Zkoušení se získá počet studentů, pro které se test nebo zkoušení připravovalo, a kdo zkoušení realizuje. Z těchto informací se již mohou vyčítat započitatelné hodiny za přípravu. Pro výpočet dalších započitatelných hodin je nutné navázat entitu Hodnocení. Počty studentů a informace o tom, kdo hodnotil zkoušení, se poté musí získat z těchto entit hodnocení, protože vykázané se mohou jen zkoušky, které se skutečně hodnotily, a také proto, že ten, kdo zkoušku vytvořil a nachystal, ji poté nemusel opravovat.

Získávání informací o semestrálních zkoušení je obdobné, jako u půlsestrálních zkoušek, s tou změnou, že typ zde nese jen informaci, zda se jedná o semestrální zkoušku,

ale i o tom, zda je zkouška ústní, písemná nebo kombinovaná, a případně o kterou část kombinované zkoušky se jedná.

Pro vykazování zkoušení doktoranda bohužel nestačí pouhé vyfiltrování předmětu podle typu studia, protože doktorand se může objevit jako student i v předmětu, který je vypsán pro jiný typ studia. Z tohoto důvodu se jako vhodné řešení jeví nalezení doktorandů pomocí datových struktur znázorněných v diagramu 5.6, a následně nalézt průnik mezi těmito záznamy a hodnoceními semestrálních zkoušek. Průnik těchto záznamů se poté vykáže jako zkoušení doktoranda a zbytek jako běžná semestrální zkouška.

Poslední tabulky 3.6 a 3.7 obsahují informace o výpočtech započítatelných hodin spojených s činnostmi, které se věnují projektovým praxím, semestrálním pracím a závěrečným pracím. Tyto tři typy činností sdílí databázovou strukturu reprezentující komisi.

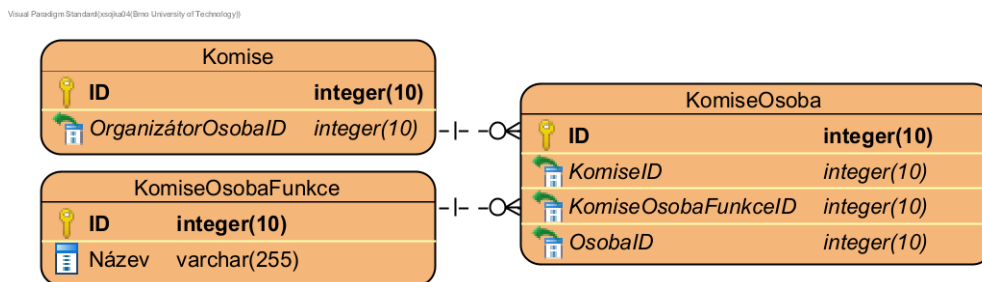


Obrázek 5.8: ERD popisující způsob uložení informací spojených s projektovými praxemi.

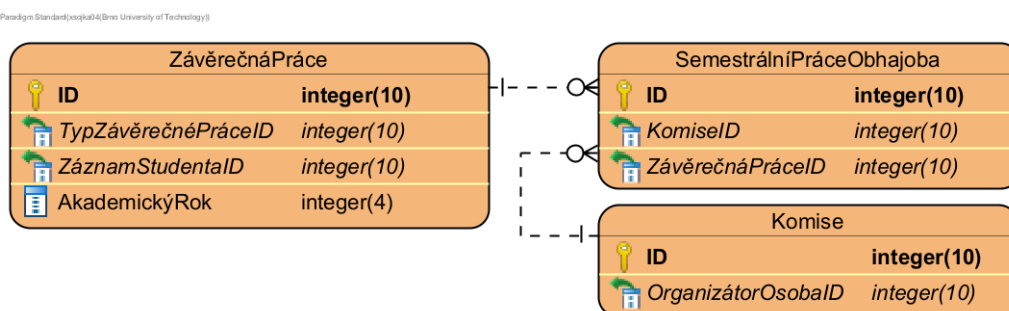
V diagramu 5.8 je ukázáno, jak jsou projektové praxe uloženy v databázi. Projektové praxe jsou uloženy v entitě Praxe a lze je dohledat pomocí identifikátoru reálného předmětu. V této entitě je dále uložena informace o tom, ke které fakultě se praxe váže, kdo je vedoucím praxe a v jakém stavu se praxe aktuálně nachází. Stav praxe lze použít pro vyfiltrování praxí takových, které lze dále použít pro výpočet započítatelných hodin. Praxe se dále váže na registraci, která obsahuje vazbu na záznam studenta, která se váže na osobu studenta. Přes entitu PraxeReálnýPředmět a vazební entitu PraxeObhajoba lze získat entitu Komise, pomocí které lze získat zbylé potřebné informace.

Identifikátor osoby organizátora komise je uložen přímo v entitě komise. Pro získání členů komise je zapotřebí využít diagramu 5.9 a navázat entitu Komise na entitu Komise-Osoba. Osoby spojené s komisí je pak dále třeba vyfiltrovat podle jejich funkce. Číselník funkcí osoby v komisi je v databázi uložen pomocí entity KomiseOsobaFunkce.

Při vykazování semestrálních prací je v první řadě potřeba přiřadit správné závěrečné práce ke správnému reálnému předmětu. Bohužel v databázi neexistuje přímá vazba mezi reálným předmětem a závěrečnou prací, a proto bude zapotřebí provést sérii kroků, kterou se vybere správný reálný předmět pro vybrané závěrečné práce. Prvním krokem je vyfiltrování závěrečných prací podle akademického roku reálného předmětu a podle typu závěrečné práce (bakalářská, diplomová nebo disertační), který odpovídá danému předmětu. Nyní se využije předpokladu, že student nebude vypracovávat v jednom akademickém roce více jak jednu



Obrázek 5.9: ERD popisující způsob uložení informací spojených s komisemi pro projektové práce a semestrální a závěrečné práce.



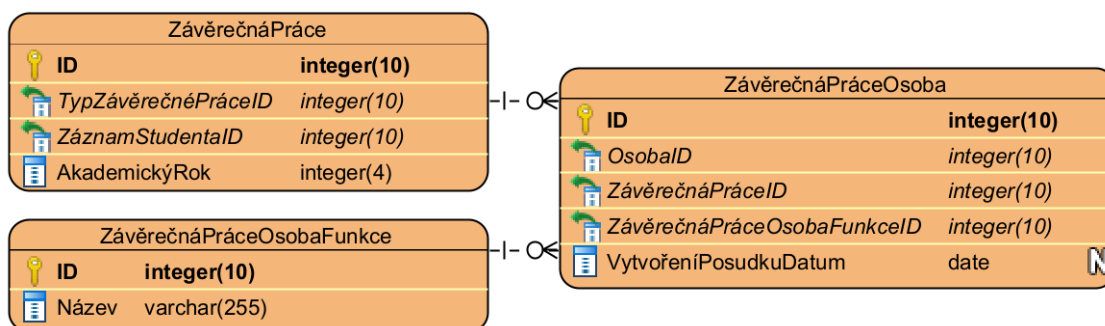
Obrázek 5.10: ERD popisující způsob uložení informací o semestrální práci a její vazbě na komisi.

semestrální práci, protože dalším krokem je navázání závěrečné práce na záznam studenta, který lze navázat na entitu HodnoceníReálnéhoPředmětu (diagram 5.5), která obsahuje identifikátor reálného předmětu. Protože reálný předmět je vázán na akademický rok, tak získané záznamy budou obsahovat jen ty práce, jejichž tvůrci měli ve stejném akademickém roce zapsaný daný reálný předmět, což je správně, protože zapsaný předmět je současně podmínkou pro přidělení započítatelných hodin.

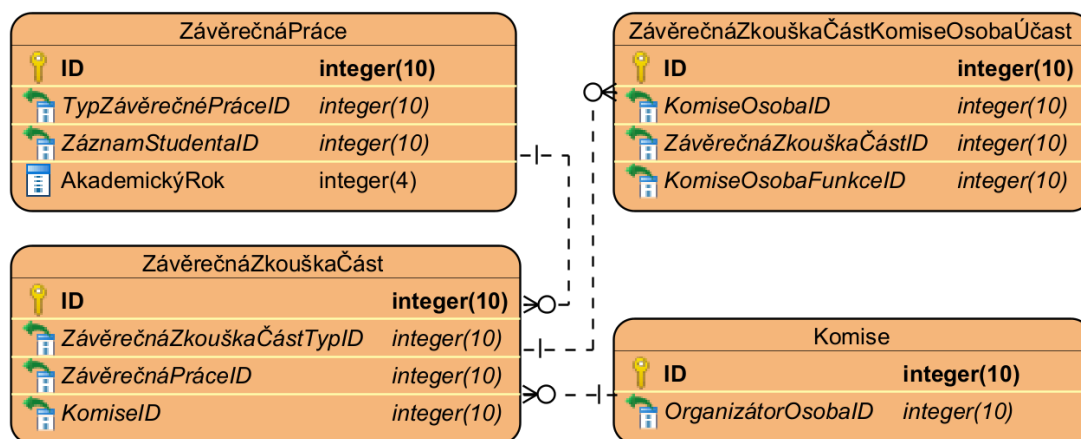
Navázání na komisi související se semestrální prací se provede pomocí vazební entity SemestrálníPráceObhajoba. Informace o členech komise se získají obdobným způsobem, jako u projektové praxe. Pro získání informace o vedoucím semestrální práce lze využít diagram 5.11. Podobným způsobem, jako u komisí, se zde na závěrečnou práci navážou osoby a následně lze specifikováním funkce osoby vybrat ze záznamů vedoucího práce.

Pro vykazování úvazků spojených s činnostmi spojenými se závěrečnými zkouškami a obhajobami závěrečných prací je zapotřebí získat záznamy závěrečných prací navázaných na reálný předmět obdobně jako u semestrálních prací. Jediným rozdílem bude, že předmět, pomocí kterého se vykazují činnosti spojené se semestrálními pracemi, se bude lišit od toho, který se používá pro vykazování činností spojených se závěrečnými zkouškami a obhajobami závěrečných prací.

Stejným způsobem jako u semestrální práce lze u závěrečné práce získat informaci o tom, kdo byl vedoucím a oponentem práce. V entitě ZávěrečnáPráceOsoba (diagram 5.11) je možné zjistit, zda byl vytvořen posudek, pomocí atributu popisujícího datum vytvoření posudku. Tímto si lze ověřit, že oponentní posudek, za který se vykazují započítatelné hodiny, byl opravdu vytvořen.



Obrázek 5.11: ERD popisující způsob uložení informací spojených se závěrečnou prací a osobami, které se k ní vážou.



Obrázek 5.12: ERD popisující způsob uložení informací spojených se závěrečnou prací a jejím navázáním na státní závěrečnou zkoušku.

V diagramu 5.12 je znázorněno, jak se závěrečná práce váže na komisi. Váže se pomocí entity ZávěrečnáZkouškaČást. Závěrečná zkouška může mít více částí a ty se liší typem, protože část zkoušky je věnována obhajobě závěrečné práce a zbylá část samostatnému zkoušení. Toto je také vidět v tabulce 3.6, kde jsou činnosti podle těchto částí rozděleny do dvou skupin.

Poslední informace, kterou je zapotřebí pro ohodnocení činností získat, je informace o tom, kdo byl v komisi přítomen u zkoušení nebo obhajoby a v jaké funkci. Přístup zde se lehce liší od přístupu, který se využíval u projektových prací a semestrálních prací. Entita KomiseOsoba zde uchovává pouze předběžný odhad toho, jaké osoby by u zkoušky měly být přítomné a jakou pozici by měly zastávat. Informace, které odpovídají tomu, kdo skutečně byl přítomen v komisi a na jaké funkci, se uchovávají v entitě ZávěrečnáZkouškaČástKomiseOsobaÚčast, která se váže na entitu ZávěrečnáZkouškaČást.

5.5 Návrh uživatelského rozhraní modulu pro správu pedagogických úvazků

Tento modul bude implementovat případy užití spojené s vyučujícím, garantem předmětu a kontrolující osobou z 5.1.

Osobní úvazky Úvazky předmětu

Akademický rok: 2022/2023 **Reálný předmět:** [zimní semestr] JP - Jazyk Python

Zobrazit

Název předmětu: Jazyk Python
Akademický rok: 2022/2023
Fakulta: Fakulta informačních technologií
Garant: Ing. Pavel Novák
Koordinátor předmětu: Ing. Pavel Novák
Ukončení: Zkouška
Počet studentů: 9

Vytvořit úvazek Vygenerovat úvazky

Vygenerované úvazky

	Typ	Druh	Osoba	Studentů	Skupin	Hodin/Skupina	Hodin
<input type="checkbox"/>	Příprava zkoušky	Automat	Ing. Pavel Novák	9	1	1	1
<input type="checkbox"/>	Hodnocení zkoušky	Automat	Ing. Jan Volný	9	1	1	1

Vybrat vše (Vybrané (0)): Přidat

Vykázané úvazky

	Připomínky	Typ	Druh	Osoba	Studentů	Skupin	Hodin/Skupina	Hodin	ZH	Zkontrolováno
<input type="checkbox"/>		Garant	Automat	Ing. Pavel Novák	8	1	1	1	10.8	✗
<input type="checkbox"/>		Dozor	Plán	Ing. Jan Volný	9	1	1	1	1.35	✓

Vybrat vše (Vybrané (0)): Smazat

Obrázek 5.13: Návrh uživatelského rozhraní stránky pro zobrazení a správu úvazků vázaných na reálný předmět.

Na obrázku 5.13 je zobrazen návrh uživatelského rozhraní, pomocí kterého se budou spravovat úvazky v rámci reálného předmětu. Ve vrchní části stránky se nachází formulář, pomocí kterého si uživatel může vybrat předmět, jehož úvazky by si chtěl zobrazit nebo je spravovat. Pod tímto formulářem se nachází základní informace o vybraném reálném předmětu.

Tlačítko „Vytvořit úvazek“ se zobrazí pouze uživatelům, kteří budou mít právo úvazky editovat. Tlačítko uživatele přesune na stránku s formulářem, pomocí kterého nový úvazek vytvoří.

Tlačítko „Vygenerovat úvazky“ uživateli zobrazí seznam automaticky vygenerovaných úvazků nadepsaný „Vygenerované úvazky“. Uživatel tyto úvazky může označit a pomocí

tlačítka „Přidat“ vykázat. Tato tlačítka a seznam budou uživateli k dispozici pouze tehdy, pokud bude mít právo úvazky editovat.

V seznamu, který je nadepsaný „Vykázané úvazky“, pak uživatel bude mít vizualizované všechny úvazky, které byly vykázány v rámci daného reálného předmětu. Přes řádek tabulky bude mít uživatel možnost se prokliknout na detail úvazku a v případě, že má právo na editaci úvazků, jej může upravit. Výjimkou je situace, kdy je vykázáný úvazek již zkontrolován. V takovém případě nesmí ani člověk s právem editace úvazek upravovat.

Pro osobu s právem kontrolovat úvazky pak bude možné u úvazku nastavit, zda je úvazek zkontrolováný či nikoliv. Toto bude moci udělat v detailu úvazku nebo přímo v tabulce nadepsané „Vykázané úvazky“.

Osobní úvazky
Úvazky předmětu

Akademický rok:
2022/2023 ▼

Zobrazit

Osoba: Ing. Jan Volný
Akademický rok: 2022/2023
ZH celkově 135.4
ZH za ak. rok: 12.75

PŘEHLED ZH PRO AKADEMCKÉ ROKY

Přidat

	Ak. rok	Norma ZH	Proplaceno ZH	Evidováno ZH	Poznámka
<input type="checkbox"/>	2021	150	0	150	
<input type="checkbox"/>	2020	170	20	190	

Vybrat vše (Vybrané (0)): **Smazat**

OSOBNÍ ÚVAZKY

Předmět	Semestr	Typ	Druh	Studentů	Skupin	Hodin/Skupina	Hodin	ZH	Zkontrolováno
GPV	Zimní	Garant	Automat	14	1	1	1	11.4	✗
JP	Zimní	Dozor	Plán	9	1	1	1	1.35	✓

Obrázek 5.14: Návrh uživatelského rozhraní stránky pro zobrazení úvazků, které byly vykázány na osobu uživatele.

Jak je znázorněno v diagramu 5.1, vyučující má možnost zobrazit všechny úvazky, které byly vykázány na jeho osobu. Návrh uživatelského rozhraní, které toto umožňuje, je zobrazen na obrázku 5.14. Ve vrchní části obrázku je formulář, pomocí kterého si uživatel může vybrat období, ze kterého by si přál zobrazit úvazky. Jednotlivé řádky tabulky nadepsané „Osobní úvazky“ budou odkazovat na detail úvazku. V tabulce nadepsané „Přehled ZH pro akademické roky“ bude mít uživatel zobrazené informace z tabulky popsané na obrázku 5.3. Právo pro správu těchto hodnot budou mít pouze ústavní a fakultní správci.

5.6 Návrh uživatelského rozhraní modulu pro zobrazení vykázaných započítatelných hodin

Pro potřeby ústavního a fakultního správce byl navržen modul, pomocí kterého se budou zobrazovat seznamy osob společně s jejich počtem vykázaných započítatelných hodin (ZH). Na obrázku 5.15 je návrh uživatelského rozhraní hlavní stránky tohoto modulu.

Pomocí formuláře zde bude moci uživatel zvolit, v závislosti na kterém akademickém roku se mu mají uživatelé zobrazit. Pro ústavního správce budou možnosti výběru univerzitního celku ve formuláři omezeny pouze na jeho ústav.

Kliknutím na položku v seznamu se pak uživatel dostane na stránku podobnou 5.14, která bude zobrazovat informace o vykázaných úvazcích a započítatelných hodinách zvolené osoby.

Akademický rok: 2022/2023 ▼ **Univerzitní celek:** UITS ▼

Zobrazit

Akademický rok: 2022/2023
Univerzitní celek: UITS

Osoba	Zařazení	ZH celkově	ZH za akademický rok
Ing. Pavel Novák	UITS, CVT	453	94
Ing. Jan Volný	UITS, CVT, SDIT	135.4	12.75

Exportovat

Obrázek 5.15: Návrh uživatelského rozhraní stránky pro zobrazení započítatelných hodin (ZH), které byly vykázány na osobu uživatele.

Kapitola 6

Implementace

V této kapitole jsou popsány jednotlivé části implementace modulu. V první části je popsáno, jakou adresářovou strukturu má projekt, následuje popis toho, jak se v modulu nakládá s právy a jak se v modulu používá parametrizace, a ve zbytku kapitoly se popisují vybrané části implementace.

6.1 Struktura

V repozitáři, do kterého jsem implementoval nový modul, existuje složka pro většinu aplikací vyvíjených v rámci IS VUT Web (StudIS, Teacher, Intraportál, E-přihláška). Pro sdílenou implementaci zde existuje ještě složka `_base`. Ve složce `_base\lib` jsou specifikované knihovny, které aplikace využívají. Pro správu těchto závislostí se používá nástroj `Composer`. Ve složce `_base\site_lib` jsou umístěny modely modelové vrstvy, která se používá pro zobrazování a manipulaci dat v databázi.

Mnou implementovaný modul se vytvářel do aplikace Teacher, která je umístěná ve složce `teacher2`. Mnou vytvořený modul má většinu své implementace umístěnou v následujících složkách:

- `teacher2\application\controllers` – V této složce je umístěný soubor `PredmetUvazkyController.php`, který obsluhuje všechny dotazy týkající se mnou vytvořeného modulu.
- `teacher2\application\forms\Predmet` – V této složce je umístěná složka `Uvazky`, která obsahuje soubory definující formuláře.
- `teacher2\application\views\scripts` – V této složce je umístěna složka `predmet-uvazky`, která obsahuje soubory se šablonami (`phtml`), pomocí kterých je definováno uživatelské rozhraní.
- `teacher2\library\Teacher\Datagrid` – Zde je umístěná složka `ZapocitatelneHodiny`, která obsahuje soubory definující datagridy, což jsou interně vyvinuté komponenty, pomocí kterých se v aplikaci Teacher vizualizují data v tabulkách.

6.2 Práva

Pro správu přístupu k datům používám dvě techniky. První z nich ověřuje práva pomocí práv, která se přiřazují v systému IS VUT Apollo. Tento přístup je vhodný, pokud přístup k datům má mít jen málo lidí a právo jim má být manuálně přiděleno, nebo pokud má mít přístup k datům jasně definovaná skupina lidí. Druhý přístup je založený na datech. Pro vyhodnocení se totiž využívá dotaz na databázi, který vrací identifikátory uživatelů, kteří mají přístup povolen. Druhý přístup je vhodný, pokud přístup závisí na datech, která se v databázi vyskytují.

Právo pro zobrazení úvazků u předmětu se získává na základě databázových dat. Stejně tak informace o tom, zda uživatel má právo u předmětu editovat informace o úvazcích. Právo pro kontrolu vykázaných úvazků se získává prvním ze zmíněných způsobů a přiděluje se manuálně.

Právo pro přístup k přehledům započítatelných hodin se získává z práv nastavených v IS VUT Apollo. Požadavkem bylo, aby si přehled vykázaných hodin mohli zobrazit vedoucí ústavu. Této skupině uživatelů bylo tedy přiřazeno právo k přístupu pomocí profese, kterou sdílí. Osobám, které nesdílí tuto profesi, se muselo právo nastavit manuálně.

Přístup k přehledu vykázaných úvazků vůči jedné osobě má osoba samotná a uživatelé, kteří mají přístup k přehledu započítatelných hodin. Z tohoto důvodu jsem použil stejný postup pro kontrolu práv.

6.3 Parametrizace

Pro správnou funkcionalitu generování a kontroly bylo zapotřebí extrahovat některé hodnoty z kódu aplikace a nabídnout je správci fakulty k editaci pomocí parametrizace, aby je mohl v případě změny na fakultě upravit. Příkladem takových hodnot mohou být hodnoty jako maximální možný počet vykazatelných příprav na semestrální zkoušku, maximální počet vykazatelných kruhových konzultací nebo konstanty pro krácení započítatelných hodin za vedení doktoranda.

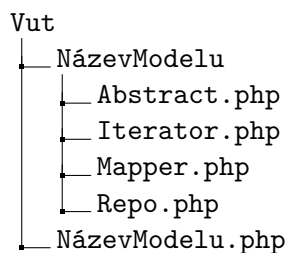
Druhou skupinou parametrů jsou parametry obsahující identifikátory předmětů. Tyto předměty mohou být buď klasické a nebo virtuální. Virtuální předměty jsou takové, které existují převážně proto, aby bylo kde vykazovat provedené činnosti. Identifikátory předmětů se pak používají pro identifikaci předmětu, ke kterému mají být činnosti vykázané. Například pro činnosti spojené s přijímacím řízením na bakalářské studium existuje virtuální předmět. Ovšem vykazované činnosti nemají v databázi žádnou existující vazbu na tento předmět a proto je parametrizace pro tento předmět nutností.

Podobný problém existuje při generování úvazků za projektovou praxi. V databázi sice existuje způsob jak rozpoznat předmět zabývající se projektovou praxí, ale neexistuje způsob, jak identifikovat ty, které jsou spojeny s programem Erasmus+. Z tohoto důvodu jsem byl nucen extrahovat identifikátory všech předmětů, kde se ohodnocení lišilo na základě toho, jestli je předmět spojený s programem Erasmus+.

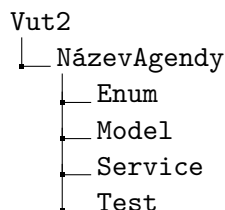
6.4 Modelová vrstva

Tato vrstva existuje v systému ve dvou verzích. První verze modelové vrstvy se nazývá „Vut1“. Modely této verze jsou umístěny v adresáři `_base\site_lib\Vut`. V této složce jsou modely do sebe navzájem zanořeny podle agendy, které se věnují. Struktura modelu

Vut1 je znázorněna na obrázku 6.1. Soubory, které jsou ve složce pojmenované po modelu, dědí z obecného modelu Vut1. Soubor `Abstract.php` je abstraktní třídou, která obsahuje atributy modelu a jejich metody typu *get* a *set*. Soubor `Iterator.php` obsahuje definice metod, které umí pracovat s polem těchto modelů. Soubor `Mapper.php` obsahuje metody, které provádí dotazy na databázi a mapují získaná data na model. V souboru `Repo.php` se nachází statické metody, které provolávají metody ze souboru `Mapper.php` a získaná data ukládají do dočasného úložiště pro případné urychlení operace při opakovaném volání metody. Soubor `NázevModelu.php` obsahuje stejnojmennou třídu a dědí ze souboru `Abstract.php`. V tomto souboru má programátor možnost vytvořit pomocné metody pro daný model.

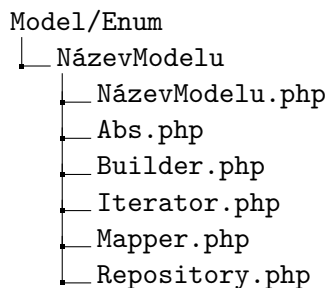


Obrázek 6.1: Struktura modelu Vut1.



Obrázek 6.2: Struktura agendy modelu Vut2.

Modely Vut2 jsou uspořádány ve složce Vut2 do složek podle agend IS. Každá agenda pak má strukturu, která je popsána na obrázku 6.2. Složky `Enum` a `Model` obsahují modely Vut2. Rozdíl mezi těmito složkami je ten, že do složky `Enum` se umísťují výhradně modely entit databázových číselníků. Složka `Service` obsahuje třídy servisní vrstvy agendy. V těchto třídách se spravuje byznys logika mezi více modely Vut2. Složka `Test` je vyhrazena pro soubory pro otestování agendy.



Obrázek 6.3: Struktura modelu Vut2.

Model Vut2 (6.3) je svou podobou velmi podobný modelu Vut1. Jedna z motivací pro tuto podobu modelů je nutnost jednoduchého přepisu modelů do nové verze. Ve verzi Vut2

jsou všechny soubory modelu uloženy v jednom adresáři a oproti staré verzi přibyl soubor `Builder.php`. V tomto souboru lze definovat, jak se bude model a iterátor instanciovat. Programátor zde může taktéž definovat závislosti na repositářích jiných modelů pomocí vkládání závislostí. Poslední velkou změnou oproti první verzi je, že v souboru `Repository.php` metody nejsou statické a programátor nyní již při volání databáze potřebuje získat instanci repositáře modelu.

Vzhledem k tomu, že v rámci mé závěrečné práce bylo nutné vytvořit moduly pro vykazování a kontrolu započítatelných hodin v základní funkční podobě velmi brzy, jsem byl nucen v rámci první verze modulu vytvořit modely pro práci s databází ve verzi Vut1. Přepis do nové verze byl časově náročný a vyžadoval relativně dost úprav kódu. Hlavním důvodem pro úpravu kódu byly závislosti na jiných repositářích ze staré verze. Závislost na staré verzi repositářů není v nových modelech povolena a z tohoto důvodu bylo zapotřebí tyto repositáře buď najít v nové modelové vrstvě, vytvořit je v nové modelové vrstvě nebo data poskytovaná repositářem dostat do modelu jako parametr. Poslední možností jsem se pokoušel vyhýbat, ale pro některé případy to bylo nezbytné. Například závislost na termínech zkoušky jsem musel vyřešit posledním zmíněným způsobem, protože její reimplementace by zabrala příliš mnoho času. Vzhledem k množství úprav jsem byl nucen sledovat, zda implementace v nové verzi vrací stále stejná data. Z tohoto důvodu jsem byl nucen implementovat testy pro identifikaci rozdílů mezi těmito dvěma verzemi (viz podkapitola 7.1).

6.5 Správa úvazků

V modelové vrstvě ve verzi Vut2 se pro práci s pedagogickými úvazky používá abstraktní model `Úvazek`. Z tohoto modelu dědí modely `Realny` a `Generovany`. Model `Realny` se používá pro práci s pedagogickými úvazky, které byly uloženy uživatelem do databáze, zatímco druhý model se používá pro práci s úvazky, které vygeneroval systém. Pro generování úvazků se používají modely, které dále specializují model `Generovany`. Tyto modely mohou generovat nové úvazky na základě dat, která si dokáží získat z databáze samy. Příkladem takového modelu je například model pro generování úvazků za přednášky. Tento model na základě znalosti identifikátoru předmětu dokáže velmi jednoduše získat z databáze všechny potřebné informace, které potřebuje pro vygenerování pedagogického úvazku. V případě, že získání informací pro generování pedagogického úvazku je příliš komplikované, jsou mu informace pro generování předány z modelu, který se specializuje na entitu, ze které je úvazek generovaný. Například při generování úvazků za vedení doktoranda jsem byl nucen vytvořit pomocnou entitu `SkolitelDoktoranda`. Hlavním důvodem pro její tvorbu byl fakt, že bylo zapotřebí dopočítat informace o tom, v jakém poměru byl doktorand veden, v případě, že měl více školitelů.

Pro generování odhadů pedagogických úvazků byly použity převážně modely popsané v předchozím odstavci. Protože se ale generování odhadů používá před uskutečněním semestru, tak úvazky, jako například úvazky za přípravu a opravu semestrální zkoušky, nelze spolehlivě vygenerovat. Tyto a podobné typy úvazků se proto odhadují tak, že se ze systému získají informace o těchto úvazcích z minulého roku a počet studentů se upraví v poměru s počtem studentů v aktuálním akademickém roce.

Generování pedagogických úvazků pro specifikovaný předmět se provádí v modelu Vut2 ve složce `Service`. Úvazky se generují pomocí definovaných dvojic typu generování a typu úvazku, které jsou pro fakultu definovány. V jednotlivých typech generování je pak definováno, které modely ze složky `Model` se mají pro generování úvazků použít.

6.6 Kontroly








System kontroluje úvazky dvěma základními způsoby. První způsob kontroluje, zda úvazky vykázané uživatelem neporušují nějaké z definovaných pravidel. Například v rámci předmětu zakončeného zkouškou by neměla být zároveň vykázána půlsestrální zkouška a průběžný test. Taktéž by v rámci jednoho předmětu neměl být vykázan úvazek za vedení řízeného studia a úvazek za přednášku. V případě, že se v rámci předmětu zjistí přítomnost nějakého nedostatku pomocí tohoto typu kontroly, tak se pomocí interního nástroje `Vut_Message` všechny nedostatky vypíší do horní části stránky.

Druhý způsob kontroly porovnává úvazky vložené do systému uživatelem s těmi, které systém vygeneroval. Vizualizace výsledků kontroly se uživateli vkládá do tabulky k jednotlivým záznamům. V případě, že úvazek kontrolován není, protože jeho typ IS neumí generovat, tak se uživateli nic o kontrole u záznamu nezobrazuje. V případě, že systém úvazek generovat umí a kontrola proběhne v pořádku, tak se u daného úvazku uživateli zobrazí vizuální potvrzení toho, že kontrola proběhla a byla úspěšná.

Porovnání uživatelem vložených úvazků a těch systémem vygenerovaných provádí metoda, která rozděluje generované úvazky do 3 skupin.

1. Existující – Skupina generovaných úvazků takových, pro které v uživatelem vložených úvazcích existuje pro jednotlivé úvazky úvazek se stejnou osobou, typem úvazku, počtem hodin a případně počtem studentů.
2. Neexistující – Skupina generovaných úvazků takových, pro které v uživatelem vložených úvazcích neexistuje pro jednotlivé úvazky úvazek se stejnou osobou, typem úvazku, počtem hodin a případně počtem studentů.
3. Neodpovídající – Skupina uživatelem vložených úvazků takových, pro které v generovaných úvazcích neexistuje pro jednotlivé úvazky úvazek se stejným počtem hodin a studentů. Počet studentů se zde řeší jen v případě, že pro daný typ úvazku je důležité kontrolovat počet studentů.

Na základě těchto tří skupin úvazků pak lze k jednotlivým záznamům v tabulce vizualizovat částečnou informaci o výsledku kontroly. Na obrázku 6.4 jsou ukázány ikonky, které systém používá pro vizualizaci. Zelená ikonka uživateli ukazuje, že kontrola proběhla a systém neidentifikoval žádnou chybu. Červená ikonka na obrázku 6.4 uživateli značí, že pro kombinaci osoby, typu úvazku, počtu hodin a studentů neexistuje generovaný ekvivalent a tudíž se zde nejspíš nachází chyba.

<input type="checkbox"/>	  	cvičení numerické (od 2019)
<input type="checkbox"/>		cvičení numerické (od 2019)
<input type="checkbox"/>	 	příprava půsem. zkoušky písemné
<input type="checkbox"/>		přednáška

Obrázek 6.4: Vizualizace kontroly uživatelem vložených pedagogických úvazků.

Pro zobrazení oranžových a modrých ikon se využívá ještě informace o tom, zda v rámci předmětu odpovídá počet vykázaných studentohodin pro daný typ úvazku. Studentohodina je pojem, který používám pro označení 1 hodiny, která je v rámci pedagogického úvazku věnována jednomu studentovi. Studentohodiny pro daný úvazek počítám jako

součin počtu hodin a počtu studentů. Pokud počet studentohodin v uživateli vložených úvazcích nesedí s těmi generovanými, pak se uživateli zobrazí oranžová ikonka s hodinami, která mu sdělí, o kolik počet vykázaných hodin pro danou kombinaci osoby a typu úvazku neodpovídá vygenerovanému. V případě, že se v systému najde v seznamu neexistujících úvazků nějaký se stejnou kombinací, tak se uživateli nabídne po najetí na oranžovou ikonku s obrázkem dokumentů. V případě, že se počet studentohodin v rámci typu úvazku shoduje, pak se uživateli zobrazí pouze modré ikonky jak je zobrazeno na obrázku 6.4. Zobrazení těchto modrých úvazků vzniklo proto, aby kontrolující osoba viděla, že v rámci předmětu nebylo vykázáno více studentohodin než systém generoval, i když systém tyto studentohodiny vygeneroval vůči jiným vyučujícím. Toto se typicky stává, když chce garant vykázat úvazky za opravování semestrální zkoušky všem, kteří se na opravování podíleli, ale do systému výsledky zkoušky zadával pouze jeden vyučující. V případě, že se zjistí, že některý z typů úvazků má v systému jiné množství studentohodin, než je vygenerováno, je uživatel na tuto skutečnost také upozorněn varovnou zprávou pomocí nástroje `Vut_Message`.

Implementace obou typů kontroly je umístěna v modelu `Vut2` ve složce `Service`. Implementace je umístěna zde, protože pro provedení kontroly je zapotřebí využít vícero typů modelů ze složek `Model` a `Enum`.

6.7 Export započítatelných hodin pro výpočet platů

V podkapitole 3.6 je popsán požadavek na export započítatelných hodin. Na základě diskuze s děkanem fakulty jsem se rozhodl data exportovat do souboru typu `XLSX`, se kterým lze pracovat pomocí nástroje `Excel`¹. Pro export dat do souboru typu `XLSX` se v rámci IS VUT Web používá knihovna `PhpSpreadsheet`² ve verzi 1.17. Data jsem se rozhodl exportovat po jednom akademickém roce a pro rozdělení dat jsem se rozhodl použít jednotlivé listy dokumentu. Započítatelné hodiny se vykazují zvlášť vůči ústavu a vůči osobě. Tato data je ještě zapotřebí rozdělit na letní a zimní semestr. Z tohoto důvodu se v exportovaném dokumentu nachází 4 listy, kde 2 se věnují osobám na fakultě a 2 listy se věnují ústavům.

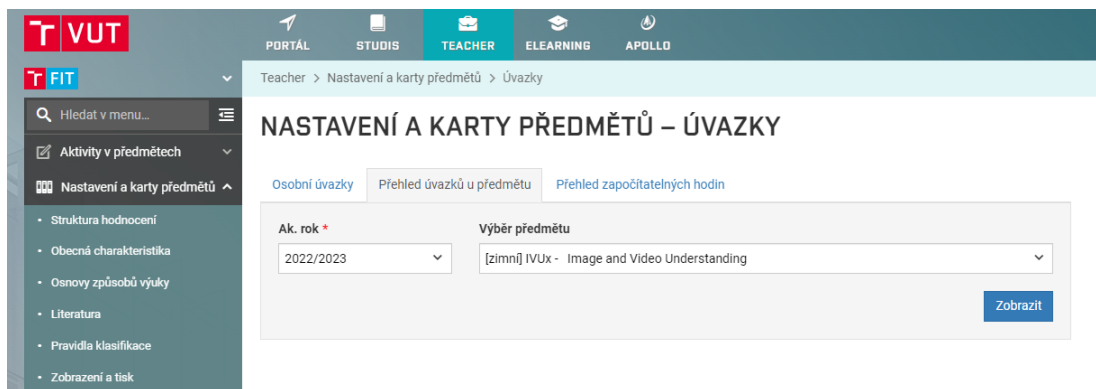
6.8 Sloučení modulů

V rámci této práce bylo mým původním záměrem vytvořit dva moduly. Jeden pro vykazování a správu pedagogických úvazků a jeden pro práci s vykázanými započítatelnými hodinami. Zjistil jsem ovšem, že v rámci vývoje aplikace `Teacher` se pro zjednodušení uživatelského rozhraní autoři snaží minimalizovat množství modulů, které se uživateli zobrazují v boční nabídce. Namísto rozdělení agendy do modulů se proto autoři snaží implementovat vícero funkcionalit do jednoho modulu. Tyto funkcionality modulu se pak od sebe oddělují pomocí záložek v horní části stránky.

Z tohoto důvodu jsem byl nucen tyto dva moduly sloučit a oddělit je pomocí zmíněných záložek. Rozdělení modulu pomocí záložek je ukázáno na obrázku 6.5. První záložka uživateli zpřístupní pedagogické úvazky vykázané vůči jeho osobě. Druhá záložka uživateli zpřístupní formulář, pomocí kterého si může zobrazit přehled vykázaných pedagogických úvazků v rámci předmětu, ke kterému má přístup. Poslední záložka není přístupná pro většinu uživatelů IS a uživateli zobrazuje souhrnné informace o vykázaných započítatelných

¹<https://www.microsoft.com/cs-cz/microsoft-365/excel>

²<https://github.com/PHPOffice/PhpSpreadsheet>

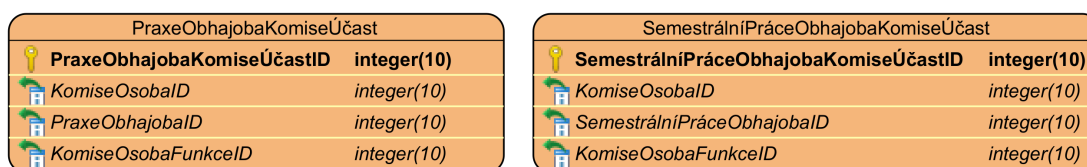


Obrázek 6.5: Databázové struktury vytvořené pro .

hodinách vůči jednotlivým osobám na fakultě nebo zvolenému ústavu fakulty. Pomocí této záložky lze exportovat data za zvolený akademický rok.

6.9 Generování pedagogických úvazků za semestrální práce a projektové praxe

Návrh generování úvazků za účast na obhajobě projektové praxe (viz diagram 5.8) vycházel z předpokladu, že členové komise se během její činnosti nebudou měnit. Tento předpoklad se bohužel ukázal jako mylný a bylo zapotřebí implementovat tuto činnost jiným způsobem. Bylo rozhodnuto, že se vytvoří nová databázová entita PraxeObhajobaKomiseÚčast (viz diagram 6.6), do které se budou evidovat záznamy o účastech členů komise u jednotlivých obhajob. Tento návrh řešení byl odvozen z řešení stejného problému u komisí spojených se závěrečnými pracemi na diagramu 5.12), kde se účast členů eviduje v entitě ZávěrečnáZkouškaČástKomiseOsobaÚčast. Protože poslední entita, která se v databázi váže na entitu Komise, je SemestrálníPráceObhajoba (diagram 5.10), bylo rozhodnuto, že i pro tento případ bude vytvořena tabulka pro evidenci účasti členů komise (viz diagram 6.6), aby se v databázi zachovala podobnost při práci s komisemi.



Obrázek 6.6: Entitně vztahový diagram entit pro evidenci účasti členů komisí projektových prací a semestrálních prací.

Kapitola 7

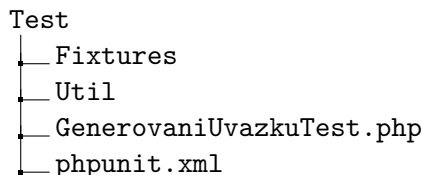
Testování

Následující kapitola se věnuje testování implementovaného modulu. Pro modelovou vrstvu modulu byly napsány testy, aby se zajistila konzistence úvazků generovaných systémem v průběhu vývoje modulu. Uživatelské rozhraní bylo testováno pomocí nástroje pro testování identit a proběhlo uživatelské testování, aby bylo odhaleno co nejvíce chyb a nedostatků, které modul obsahoval a mohly být opraveny.

7.1 Testování správnosti generování pedagogických úvazků

Pro zajištění konzistentního vývoje a budoucí udržovatelnosti modulu bylo zapotřebí vyvinout nástroj, který otestuje, že IS generuje pedagogické úvazky správně. Z tohoto důvodu jsem se rozhodl vytvořit nástroj, pomocí kterého se získají již zkontrolované pedagogické úvazky z informačního systému a uloží se do textové podoby, aby bylo možné je manuálně editovat. Takto uložená data se budou následně používat jako referenční při testování generování pedagogických úvazků.

Bylo rozhodnuto, že pro tento typ testování se bude používat nástroj PHPUnit¹, který se v IS VUT Web používá ve verzi 8.5. Tento typ testování je v rámci IS VUT Web nový a proto musím pro referenci dat používat soubor namísto například testovací databáze. Soubory pro testování systému jsou umístěny ve složce **Test** (7.1). V této složce se nachází 2 podsložky. Podsložka **Fixtures**, ve které jsou uložena referenční data, vůči kterým se testy provádí. Podsložka **Util** obsahuje nástroje pro tvorbu testovacích modelů pedagogických úvazků a jejich porovnávání. V souboru **GenerovaniUvazkuTest.php** se nachází jednotlivé případy pro otestování a v souboru **phpunit.xml** je umístěna konfigurace testů.



Obrázek 7.1: Struktura složky **Test**, která se vyskytuje v agendě Vut2 spravující problematiku pedagogických úvazků a započítatelných hodin.

V rámci této práce jsem tyto testy využíval pro kontrolu poté, co jsem v generování pedagogických úvazku v modelu ve verzi Vut2 udělal úpravy a při přechodu z verze Vut1

¹<https://phpunit.de/>

na verzi Vut2. Protože jsem se rozhodl pro testování použít speciální modely, tak úprava pro práci s jinou verzí modelu byla velmi jednoduchá, protože jediné, co bylo zapotřebí upravit, byla metoda pro nastavení atributů modelu.

7.2 Testování pomocí testování identit

V IS VUT Web existuje funkcionální testování identit, která vývojáři umožňuje částečně si vypůjčit identitu uživatele a otestovat s ní jím upravovanou funkcionální. Toto je velmi užitečná funkce, protože umožňuje vývojáři otestovat jak práva, kterými by měl uživatel v systému disponovat, tak zobrazování dat v daném prostředí. Z tohoto důvodu jsem se vždy pokoušel otestovat funkcionální všech typů uživatelů v mnou vyvíjených modulech pomocí této funkcionality před tím, než byly nové funkcionality určeny k nasazení.

7.3 Uživatelské testování

Pro získání zpětné vazby k mým modulům a informací o nalezených chybách jsem využil vícero kanálů. Testující uživatelé mě mohli informovat o nedostacích modulu pomocí aplikace Microsoft Teams², požadavkového systému IS VUT a pomocí on-line dotazníku, který se skládal z následujících otázek a podotázek:

- Vykazujete v systému úvazky?
 - Použil(a) jste systémem generované úvazky?
 - Použil(a) jste systémem odhadované úvazky?
 - Použil(a) jste připomínky/kontroly, které systém generuje k vykázaným úvazkům?
 - Měl(a) jste s něčím problémem při vykazování úvazků?
- Narazil(a) jste při zobrazování úvazků na problém?
 - Jsou pro Vás zobrazené informace dostatečné?
- Používáte IS pro zobrazení přehledu započítatelných hodin fakulty nebo ústavu?
 - Přijdou Vám zobrazené informace dostačující?
 - Přijdou Vám informace v exportovaném dokumentu dostačující?
 - Máte při zobrazení osobního přehledu vykázaných hodin doktoranda k dispozici všechny informace, které potřebujete?

Požadavky pomocí požadavkového systému IS VUT jsem společně s připomínkami přes aplikaci Microsoft Teams přijímal po většinu doby vývoje modulu. Dotazník pro uživatelské testování byl naopak vytvořen až v závěru mé práce a distribuoval jsem ho mezi vyučující za pomoci mého vedoucího práce. Pomocí zmíněných kanálů byly reportovány některé nedostatky vícekrát a řešení některých reportovaných nedostatků bylo tak jednoduché, že jsem si je dovilil v této práci nezmiňovat. Následují nedostatky, které mi byly reportovány pomocí zmíněných kanálů a dle mého názoru bylo vhodné je v této kapitole zmínit:

1. Chybí přehled započítatelných hodin vykázaných vůči jednotlivým typům úvazků.

²<https://www.microsoft.com/cs-cz/microsoft-teams>

2. Chybí upozornění na chybějící dozory u zkoušek.
3. Negenerují se pedagogické úvazky za projekty, které jsou v systému vedené pod typem *domácí úkol*.
4. Negenerují se pedagogické úvazky za přednášky
5. Negenerují se pedagogické úvazky za obhajoby disertačních prací
6. Nefunkční tlačítko pro vkládání odhadovaných úvazků.
7. Systém tvrdí, že v předmětu proběhlo příliš mnoho půlsestrálních zkoušek, i když to není pravda.
8. Systém generuje pedagogický úvazek za demonstrační cvičení za vykázané hodiny numerického cvičení.
9. Systém generuje v pedagogických úvazcích za přijímací komise úvazky s nesmyslnými daty.

Nedostatky 1 a 2 popisují funkcionalitu, která uživatelům v modulu chyběla. Na základě konzultací jsem se dozvěděl, že pro kontrolující osobu je důležité vědět, kolik je za činnost vykázaných započítatelných hodin, protože je to ten nejdůležitější ukazatel při rychlé kontrole předmětu, protože se přímo váže na finance, které jsou pro předmět k dispozici. Z těchto důvodů jsem se rozhodl přidat do modulu přehled započítatelných hodin vykázaných vůči jednotlivým typům úvazků. Při vykreslení informace o tom, kolik započítatelných hodin bylo vůči jednotlivým typům pedagogických úvazků vykázáno, je dle mého názoru užitečné, když se vykreslí i informace o počtu hodin a studentohodin, které byly v rámci činnosti vykázané. Vyučujícím při vykazování umožní rychle a jednoduše se podívat na počet vykázaných hodin a studentohodin u daného typu úvazku a detekovat tak nepřesnost ve vykázaných úvazcích.

V rámci nedostatku 2 si uživatel stěžoval, že systém neupozorňuje na chybějící dozory u zkoušek. Toto chování je způsobeno tím, že dozory se v IS VUT neevidují a tím pádem je systém neumí vygenerovat. Rozhodl jsem se ale implementovat kontrolu, která vyučujícího upozorní na to, že v předmětu není vykázaný dozor u semestrální zkoušky. Toto upozornění se vygeneruje, pokud je v předmětu vykázaný úvazek za semestrální zkoušku. Upozornění se negeneruje v případě, že je úvazek vykázaný jen za zkoušku ústního typu, protože u tohoto typu zkoušky podle [10] nelze vykázat úvazek za dozor.

V případě nedostatku 3 si uživatel stěžoval, že mu modul negeneruje úvazek za projekt, který v rámci předmětu vedl. Ukázalo se, že v systému tento projekt byl veden ne pod typem *projekt*, ale pod typem *domácí úkol*. Na základě konzultace se rozhodlo, že modul bude nadále ignorovat tento typ, protože se nezjistilo, že by vyučujícímu poskytoval nějakou další funkcionalitu, a protože FIT VUT má zájem o sjednocení chování vyučujících v IS VUT.

Nedostatky 4 a 5 nebyly vyhodnoceny jako chyby modulu. V případě nedostatku 4 to bylo proto, že garant předmětu neměl u přednášek zaznamenáno, kdo je provede. Druhý nedostatek byl způsoben tím, že v databázi nebyla zaznamenána účast členů komise pomocí tabulky zobrazené na obrázku 5.12. Uživatele jsem o těchto skutečnostech informoval.

Chyba zmíněná v bodě 6 byla způsobena tím, že krátce před odhalením této chyby byla prováděna refaktorizace kódu, který realizoval akce spravující pedagogické úvazky vyvolané pomocí tlačítek. V rámci této refaktorizace se bohužel opomenulo otestování tohoto konkrétního tlačítka a nebyla odhalena tato chyba. Tuto chybu jsem ze systému odstranil.

Chyba 7 byla způsobena tím, že u větších předmětů se vytváří větší počet zkoušení, než jich reálně probíhá, protože se vytváří zkoušení pro každou učebnu, ve které půlsestrální zkouška probíhá. Kontrolu jsem upravil tak, že nyní nepočítám, kolik zkoušení proběhlo, ale sečtu počty studentů, kteří se jednotlivých zkoušení účastnili, a z této hodnoty si pak již dopočítám, kolikrát se půlsestrální zkouška konala.

Nedostatek popsany v 8 byl způsobený tím, že se pro demonstrační cvičení v rozvrhu nepoužíval typ, který by jej jasně specifikoval, ale používal se typ *numerické cvičení*. Bohužel toto způsobilo to, že systém nebyl schopen zároveň správně identifikovat úvazky za demonstrační cvičení a numerické cvičení. Tento problém byl vyřešen tak, že bylo rozhodnuto, že se pro demonstrační cvičení v rozvrhu bude v rámci fakulty používat exkluzivně typ *seminář*.

Nedostatek 9 popisoval situaci, kdy systém uživateli generoval úvazky vůči neexistujícím osobám. V tomto případě se ovšem bohužel nejednalo o chybu modulu, ale o chybu v datech, která již byla odhalena v průběhu vývoje modulu a bohužel nebyla do té doby opravena. V modulu jsem ponechal možnost generovat úvazky vůči neexistujícím osobám, protože uživateli dokáže alespoň napovědět, že něco není v pořádku a že by měl vykazování těchto úvazků věnovat zvláštní pozornost.

Kapitola 8

Závěr

Cílem této práce bylo prostudovat agendu vykazování a kontroly pedagogických úvazků, vytvořit návrh modulů do IS VUT, které jsou zapotřebí pro to, aby se agenda mohla přemístit do nového informačního systému, a navržené řešení implementovat.

Prostudoval jsem to, jak se agenda řešila v informačním systému Fakulty informačních technologií VUT, a zjistil jsem, co je zapotřebí implementovat do IS VUT, aby bylo možné agendu spravovat z tohoto informačního systému. Prostudoval jsem IS VUT a technologie, pomocí kterých byl implementován, se zaměřením na ty části systému, které souvisí s řešenou agendou. V rámci práce jsem vypracoval návrh modulů, které je zapotřebí do IS VUT vytvořit. Tento návrh jsem použil pro implementaci modulu do IS VUT a mnou implementovaný modul jsem následně otestoval.

V mnou implementovaném modulu je možné vykazovat pedagogické úvazky manuálně, generovat je z dat, která již byla do IS zadána v jiném modulu, a nebo je vykazovat jako odhady na základě dat z minulých let. Modul uživateli pomáhá při kontrole vykázaných úvazků porovnáváním vykázaných úvazků s daty o činnostech, kterými IS disponuje, a kontrolou limitů a omezení pedagogických úvazků v rámci kontrolovaného předmětu. Modul uživateli taktéž umožňuje zobrazit přehled pedagogických úvazků, které byly vykázány vůči jeho osobě, a vědeckému oddělení a vedení fakulty poskytuje nástroje pro zobrazení přehledů úvazků.

V rámci mé práce jsem se naučil, co obnáší práce na dlouhodobém projektu jako je IS VUT. Naučil jsem se, že práce na projektu, který se vyvíjí a udržuje už téměř 20 let, se podstatně liší od práce na projektu menšího charakteru a tým vývojářů nutí uvažovat v úplně jiném časovém horizontu.

První nasazení modulu do IS VUT proběhlo již v září akademického roku 2022/2023. V květnu 2023 se pomocí něj již vytvořilo více než 4000 pedagogických úvazků pro více než 240 předmětů.

V průběhu mé práce o řešení podobného problému v IS VUT projevila zájem i Fakulta stavební VUT. Rád bych jim poskytl své znalosti k tomu, aby mohli nastavit své procesy pro vykazování pedagogických úvazků tak, aby je bylo možné v co nejkratším čase implementovat do IS VUT.

Literatura

- [1] ACHOUR, M., BETZ, F., DOVGAL, A. et al. *PHP: Hypertext Preprocessor* [online]. PHP Documentation Group, prosinec 2022 [cit. 2022-12-04]. Dostupné z: <https://www.php.net/manual/en/>.
- [2] ALLEN, R., BROWN, S. a LO, N. *Zend Framework in Action*. 1. vyd. Greenwich: Manning Publications, prosinec 2009. ISBN 1933988320.
- [3] CORPORATION, O. *Managing Software in Oracle Linux: Oracle Linux 8 Application Streams* [online]. Oracle Corporation, prosinec 2022 [cit. 2022-12-08]. Dostupné z: <https://docs.oracle.com/en/operating-systems/oracle-linux/software-management/sfw-mgmt-ApplicationStreamModuleLifeCycle.html>.
- [4] CORPORATION, O. *Oracle Database Documentation* [online]. Austin, Texas, USA: Oracle Corporation, prosinec 2022 [cit. 2022-12-05]. Dostupné z: <https://docs.oracle.com/en/database/oracle/oracle-database/>.
- [5] CORPORATION, O. *Oracle Real Application Clusters 19c Technical Architecture* [online]. Oracle Corporation, prosinec 2022 [cit. 2022-12-05]. Dostupné z: <https://www.oracle.com/webfolder/technetwork/tutorials/architecture-diagrams/19/rac/main.html>.
- [6] GROFF, J. a WEINBERG, P. *SQL the complete reference*. Berkeley, California: McGraw-Hill Companies, 1999 [cit. 2022-01-11]. ISBN 0-07-211845-8. Dostupné z: [http://englishonlineclub.com/pdf/SQL%20-%20The%20Complete%20Reference%20\[EnglishOnlineClub.com\].pdf](http://englishonlineclub.com/pdf/SQL%20-%20The%20Complete%20Reference%20[EnglishOnlineClub.com].pdf).
- [7] GRUDL, D. *Dokumentace Nette* [online]. Nette Foundation, prosinec 2022 [cit. 2022-12-05]. Dostupné z: <https://doc.nette.org/>.
- [8] GU, M. a TANG, K. Comparative analysis of WebForms MVC and MVP architecture. In: *2010 The 2nd Conference on Environmental Science and Information Application Technology* [online]. IEEE, 2010, sv. 2, s. 391–394. DOI: 10.1109/ESIAT.2010.5567323. Dostupné z: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5567323>.
- [9] KOČÍ, R. a ZEMČÍK, P. *Úplné znění směrnice č. Č. 13/2018 Pravidla o organizaci studia na FIT* [online]. 2018 [cit. 2022-11-12]. Dostupné z: <https://www.vut.cz/uredni-deska/vnitri-legislativa-fit/smernice-c-13-2018-pravidla-o-organizaci-studia-na-fit-d175808>.

- [10] KOČÍ, R. a ZEMČÍK, P. *Úplné znění směrnice č. 4/2019 - Instrukce pro výpočet pedagogických úvazků pro interní potřeby FIT VUT* [online]. 2019 [cit. 2022-11-12]. Dostupné z: <https://www.fit.vut.cz/fit/info/smernice/sm2019-04-uz.pdf>.
- [11] RAY, L. *Delphi v kostce: Pohotová referenční příručka*. 1. vyd. Praha: Computer Press, 2000. ISBN 80-7226-361-7.

Příloha A

Obsah přiloženého paměťového média

Na přiloženém paměťovém médiu se nachází:

- Složka `source_codes`, která obsahuje zdrojové kódy implementovaného modulu. Tato složka taktéž obsahuje své vlastní `README`, ve kterém je popsáno, jakým způsobem jsou zde soubory uloženy.
- Soubor `poster.pdf`, který obsahuje plakát popisující tuto práci.
- Soubor `poster.xcf`, který obsahuje data pro vytvoření plakátu.
- Soubor `thesis.pdf`, který obsahuje technickou zprávu určenou pro čtení na počítači.
- Soubor `thesis_print.pdf`, který obsahuje technickou zprávu určenou pro tisk.
- Složka `thesis`, která obsahuje zdrojové kódy a ostatní data potřebné pro vytvoření `thesis.pdf` a `thesis_print.pdf`.
- Soubor `README`, ve kterém je popsán obsah paměťového média.

Příloha B

Plakát

Moduly IS VUT pro vykazování a kontroly pedagogických úvazků

Cíl práce

Cílem této práce bylo prostudovat agendu vykazování a kontroly pedagogických úvazků a vytvořit návrh modulů do IS VUT, které jsou zapotřebí pro to, aby se agenda mohla přemístit do nového informačního systému. Jelikož činnosti, za které se vykazují pedagogické úvazky, jsou z velké části již evidované v IS VUT, tak se tato práce zaměřuje na maximální možnou automatizaci procesů vykazování a kontroly úvazků.

Výstup práce

Modul uživatelům spravujícím pedagogické úvazky pro daný předmět zobrazuje již vykázané úvazky, a dále jim nabízí úvazky, které odhadl a vygeneroval z dostupných dat. Vygenerované úvazky by měly být přesné a odpovídat činnostem, které byly v rámci předmětu uskutečněny. Systém by je neměl mít problém generovat, dokud vyučující předmětu budou využívat IS VUT způsobem, jakým FIT VUT zamýšlel při tvorbě tohoto modulu. Nabízené odhadované úvazky nejsou tak přesné, jako ty generované, protože se při odhadu opírají o úvazky vykázané v předchozím roce.

Vyučujícím modul zobrazuje navíc jejich osobní přehled vykázaných pedagogických úvazků.

Pro vedení fakulty a vědecké oddělení se v modulu zobrazuje přehled vykázaných započítatelných hodin. Vedení fakulty a vědecké oddělení si taktéž mohou zobrazit detail vykázaných pedagogických úvazků vyučujícího a u doktorandů spravovat normy započítatelných hodin, které musí během studia vykázat.

Výsledky

V květnu 2023 se pomocí tohoto modulu vytvořilo již více než 9100 pedagogických úvazků pro více než 700 předmětů.

Číslo předmětu	Název předmětu	Úvazky	...
123456	Matematika	120	...
789012	Fyzika	80	...
345678	Chemie	100	...
901234	Biologie	60	...
567890	Historie	40	...

Číslo předmětu	Název předmětu	Úvazky	...
123456	Matematika	120	...
789012	Fyzika	80	...
345678	Chemie	100	...
901234	Biologie	60	...
567890	Historie	40	...

Číslo předmětu	Název předmětu	Úvazky	...
123456	Matematika	120	...
789012	Fyzika	80	...
345678	Chemie	100	...
901234	Biologie	60	...
567890	Historie	40	...



Autor: Bc. Matěj Sojka
E-mail: xsojka04@stud.fit.vutbr.cz
Vedoucí práce: Ing. Jaroslav Dytrch, Ph.D.

Brno 2023